

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
Fakulta elektrotechnická

# BAKALÁŘSKÁ PRÁCE

2008

Petr Mik

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
Fakulta elektrotechnická  
Katedra řídicí techniky

**Metody fuzzy shlukové analýzy při aproximaci funkcí**

Vedoucí práce: Ing. Petr Hušek, Ph.D.

Autor: Petr Mik

Praha 2008

## Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze dne .....  
Petr Mik

## Poděkování

Na tomto místě bych rád poděkoval především Ing. Petru Huškovi, Ph.D., vedoucímu mé bakalářské práce, a také svým rodičům.

## **Anotace**

Tato práce je zaměřena na aproximaci funkcí pomocí metod fuzzy shlukové analýzy s využitím Mamdani a Takagi-Sugeno fuzzy systémů. Jsou zde uvedeny základní mechanismy nastavování parametrů těchto systémů pomocí výsledků fuzzy c-means a Gustafson-Kesselova algoritmu. Dále následuje ověření kvality aproximace na několika cvičných funkcích, a to jak jedné, tak i dvou proměnných. Při ověřování výsledků je brán ohled nejen na správnost aproximace, ale také na časovou náročnost uvedených algoritmů.

## **Annotation**

This bachelor's project is focused on the approximation of function with use algorithms and methods of fuzzy clustering analysis for recognition of data, and Mamdani or Takagi-Sugeno fuzzy systems for approximation of desired function. There are mentioned the basic concept of adjustment parameters of this systems with the aid of fuzzy c-means (FCM) and Gustafson-Kessel (GK) algorithms in this project. The verification of approximation on two or three dimensional data ensues next. The time and the quality of approximation are measured by verification of results.

## Obsah

<b>1</b>	<b>Úvod</b> .....	<b>1</b>
<b>2</b>	<b>Shluková analýza (SA)</b> .....	<b>2</b>
2.1	Metriky .....	2
2.2	Dělení shlukové analýzy.....	3
2.3	Fuzzy shluková analýza.....	4
2.4	Probabilistická fuzzy shluková analýza.....	5
2.5	Possibilistická fuzzy shluková analýza.....	5
2.6	Objektivní funkce .....	6
2.7	Základní algoritmy fuzzy shlukové analýzy.....	7
2.7.1	Algoritmus 1, probabilistická SA .....	7
2.7.2	Algoritmus 2, possibilistická SA .....	8
2.8	Fuzzy c-means algoritmus (FCM), probabilistická verze.....	9
2.9	Gustafson-Kesselův algoritmus (GKA), probabilistická verze .....	10
<b>3</b>	<b>Fuzzy systémy</b> .....	<b>12</b>
3.1	Fuzzy logické operace .....	14
3.2	Vyvozování a defuzzifikace v Mamdani systémech .....	15
3.3	Vyvozování a defuzzifikace v Takagi-Sugeno systémech .....	15
<b>4</b>	<b>Aproximace funkcí</b> .....	<b>16</b>
4.1	Aproximace funkcí pomocí algoritmu FCM .....	16
4.2	Nastavení parametrů Mamdani systému.....	17
4.3	Aproximace funkcí pomocí GK .....	18
4.4	Nastavení parametrů Takagi-Sugeno systému .....	19
4.5	Měření kvality aproximace .....	20
4.6	Aproximace funkcí jedné proměnné.....	21
4.6.1	Aproximace funkce kosinus – FCM.....	21
4.6.2	Aproximace funkce kosinus – GK.....	23
4.6.3	Aproximace aperiodické funkce – FCM .....	26
4.6.4	Aproximace aperiodické funkce – GK .....	28
4.7	Aproximace funkcí dvou proměnných .....	30
4.7.1	Aproximace sedlovité funkce - FCM .....	31
4.7.2	Aproximace sedlovité funkce - GK.....	33
4.7.3	Aproximace dvouvrcholové funkce.....	35

4.7.4	Aproximace dvouvrcholové funkce – FCM .....	36
4.7.5	Aproximace dvouvrcholové funkce – GK.....	38
4.8	Zhodnocení výsledků aproximace .....	40
<b>5</b>	<b>Závěr .....</b>	<b>41</b>

# 1 Úvod

Tato práce je zaměřena na aproximaci funkcí pomocí metod fuzzy shlukové analýzy s využitím fuzzy systémů. Při používání algoritmů shlukové analýzy bylo využito jejich implementace ve Fuzzy Clustering and Data Toolbox [5], a pro návrh fuzzy systémů pak Fuzzy toolbox, jenž je standardní součástí programu MATLAB®.

To, že je aproximace pomocí fuzzy systémů možná, nám zaručuje fuzzy aproximační teorém, který říká, že „libovolnou spojitou funkci můžeme aproximovat tak přesně, jak je zapotřebí pomocí konečné množiny fuzzy proměnných, hodnot a pravidel“ [2, str. 65]. O tom, jak to provést, aproximační teorém nic neříká. A právě cílem této práce je ukázat jeden z možných postupů.

V prvních kapitolách je provedeno rozdělení fuzzy shlukové analýzy a jsou zde popsány dva z jejích základní algoritmů: Fuzzy c-means algoritmus (FCM) a Gustafson-Kesselův algoritmus (GK). Také jsou zde probrány jejich principy, výhody a nevýhody. Dále se práce věnuje popisu Mamdani a Takagi-Sugeno fuzzy systémů. Jsou zde uvedeny hlavní rozdíly mezi těmito fuzzy systémy, tvar inferenčních pravidel, a také způsoby vyvozování. Pro úplnost je zařazeno krátké shrnutí operací fuzzy logiky. Kapitola 4 je věnována samotné aproximaci. Nejprve je popsána tvorba pravidel pro fuzzy systémy, a to jak z výsledků GK, tak z výsledků FCM. Poté se práce věnuje srovnání výsledků aproximací různých funkcí, a to jak z hlediska časové náročnosti obou algoritmů, tak z hlediska přesnosti aproximace.



## 2 Shluková analýza (SA)

Základním cílem shlukové analýzy je rozdělit zadaná data (objekty) do skupin (shluků či tříd), a to tak, aby platily dvě následující podmínky:

- Homogenita uvnitř shluku, to znamená, že data uvnitř každého jednoho shluku by si měla být co nejpodobnější.
- Heterogenita mezi shluky, což znamená co největší různost a odlišnost mezi daty patřících k odlišným shlukům.

Přítom přesnější určení pojmů podobnost a odlišnost záleží na charakteru dat. Můžeme uvažovat vzdálenost pro klasifikaci bodů v rovině či prostoru, nebo statistické veličiny (střední hodnota, rozptyl či korelace...) pro elektrické signály.

### 2.1 Metriky

Pro míru podobnosti můžeme v případě  $n$ -dimensionálního prostoru  $\mathbf{R}^n$  použít ve shodě s [4] metriku  $d(a,b)$ . To je funkce, která splňuje, pro dva body  $a,b \in \mathbf{R}^n$ , následující vlastnosti :

$$d(a,b) = 0, \text{ právě tehdy, když } a = b \text{ (identita),} \quad (2.1)$$

$$d(a,b) \geq 0, \quad (2.2)$$

$$d(a,b) = d(b,a) \text{ (symetrie),} \quad (2.3)$$

$$d(a,b) \leq d(a,c) + d(c,b), \text{ (trojúhelníková nerovnost).} \quad (2.4)$$

Takovou metrikou je například Minkowského metrika:

$$d(a,b) = \left( \sum_{i=1}^n (a_i - b_i)^k \right)^{1/k}, k = 1, 2, \dots \quad (2.5)$$

Pokud dosadíme za  $k$ , dostaneme některé známé metriky:

Pro  $k = 1$  získáme manhattanskou vzdálenost (také známou jako vzdálenost městských bloků).

$$d(a,b) = \sum_{i=1}^n |a_i - b_i| \quad (2.6)$$

Běžně používaná Eukleidovská vzdálenost vznikne dosazením za  $k = 2$ :

$$d(a,b) = \left( \sum_{i=1}^n (a_i - b_i)^2 \right)^{1/2} \quad (2.7)$$

Později budeme používat ještě jeden druh metriky - takzvanou Mahalanobisovu vzdálenost. Tato metrika má jednu důležitou vlastnost, a to, že je invariantní vůči změně měřítka. To znamená, že není závislá na rozsahu jednotlivých měřených veličin.

$$d(\mathbf{a}, \mathbf{b}) = \left( (\mathbf{a} - \mathbf{b})^T \mathbf{A}^{-1} (\mathbf{a} - \mathbf{b}) \right)^{1/2} \quad (2.8)$$

V tomto případě uvažujeme  $\mathbf{a} = (a_1, a_2, \dots, a_n)^T$ ,  $\mathbf{b} = (b_1, b_2, \dots, b_n)^T$ .  $\mathbf{A}$  je symetrická, pozitivně definitní kovarianční matice. Používáme-li tuto vzdálenost, její hodnota není závislá pouze na bodech  $\mathbf{a}$ ,  $\mathbf{b}$ , ale také na rozmístění ostatních bodů (kvůli kovarianční matici).

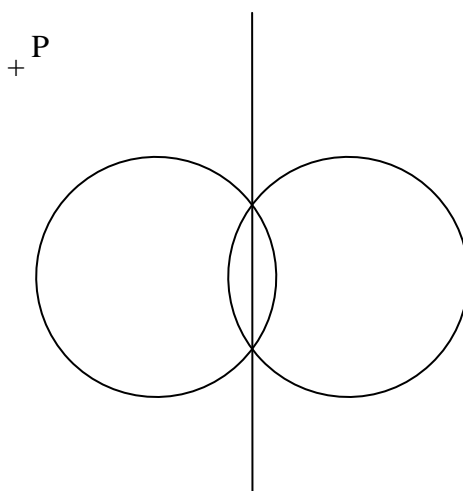
## 2.2 Dělení shlukové analýzy

Shlukovou analýzu můžeme podle [1] rozdělit do několika tříd, a to na základě postupů, které daná třída využívá:

- a) Heuristické shlukování – využívá geometrické metody, kdy vícedimenzionální objekty jsou analyzovány pomocí jejich projekce do dvou či tří dimenzionálního prostoru. Používá se zde metoda vlastních směrů PCA (Principal Component Analysis).
- b) Deterministické shlukování – provádí přiřazení objektu do právě jednoho shluku.
- c) Probabilistické shlukování – pro každý objekt je určen stupeň příslušnosti, se kterou patří ke každému shluku. Součet všech příslušností jednoho objektu je roven jedné.
- d) Possibilistické shlukování – pro každý objekt je určen stupeň příslušnosti ke každému shluku.
- e) Hierarchické shlukování – výsledkem je hierarchická struktura shluků, kdy na nejnižší stupni je přiřazen každému objektu jeden shluk a na nejvyšším stupni patří všechny objekty do jednoho shluku.
- f) Metody SA založené na objektivní funkci – jsou založené na minimalizaci chyby přiřazení daného objektu k danému shluku.

## 2.3 Fuzzy shluková analýza

Uvažujeme-li klasickou deterministickou shlukovou analýzu používající ostré množiny, pak jejím výsledkem je přiřazení daného objektu právě do jednoho shluku (prvek do ostré množiny buď patří, nebo ne). To se může zdát výhodné, ale v některých případech to není přesně to, co bychom očekávali jako správný výsledek. Například uvažujme případ dvou shluků reprezentovaných dvěma vzájemně se protínajícími kruhy (Obr 2.1). Zamyslíme-li se nad otázkou, kam přiřadit body, které mají stejnou vzdálenost od obou kruhů (těmito body je dána přímka vyznačená na obrázku), pak bychom je intuitivně přiřadili k oběma kružnicím. Klasické metody však přiřadí každý bod právě k jedné kružnici. Právě takové případy byly motivací k zavedení fuzzy množin do shlukové analýzy. S jejich použitím můžeme určit každému bodu příslušnost, se kterou patří k danému shluku. Tuto vlastnost matematicky popisuje takzvaná funkce příslušnosti  $\mu(x, k)$ , vyjadřující, jak hodně patří daný bod  $x$  ke shluku  $k$ . Tato funkce nabývá všech hodnot z intervalu  $\langle 0,1 \rangle$ . V případě klasické shlukové analýzy (s ostrými množinami) by tato funkce nabývala pouze dvou hodnot, a to  $\{0,1\}$ -objekt ke shluku patří, a nebo nepatří. Nyní se můžeme vrátit k našemu příkladu. Sporným bodům přiřadíme příslušnost  $\mu(x, k_i) = 0,5$ , a to k oběma kruhům  $k_1, k_2$ . Tento zápis znamená, že daný bod  $x$  náleží ke kruhu  $k_1$  s příslušností 0,5, a také ke kruhu  $k_2$  s příslušností 0,5. Tím jsme vyřešili problém, kdy byl bod přidělen právě do jednoho shluku, i když by bylo přirozenější jej zařadit do více shluků.



Obr. 2.1: Přímka a dva protínající se kruhy

## 2.4 Probabilistická fuzzy shluková analýza

O probabilistickém fuzzy dělení mluvíme (podle [1]), jsou-li splněny dvě následující podmínky:

$$\forall x \in X : \sum_{k \in K} \mu(x, k) = 1, \quad (2.9)$$

$$\forall k \in K : \sum_{x \in X} \mu(x, k) > 0, \quad (2.10)$$

kde  $X$  je množina všech objektů, které se snažíme klasifikovat, a  $K$  je množina všech shluků, ke kterým přiřazujeme prvky z  $X$ . První podmínka vyjadřuje určitý druh normalizace. Říká, že součet příslušností daného bodu ke všem shlukům musí být roven jedné. Druhá podmínka nám zaručuje neprázdnost každého shluku, tedy, že ke každému shluku musí náležet alespoň jeden prvek s nenulovou příslušností. Hlavním problémem probabilistického dělení je normalizace, tedy první pravidlo. Opět se vrátíme k našemu příkladu dvou protínajících se kruhů a přímkou. Všechny body na přímce mají stejnou vzdálenost od středů obou kruhů. Intuitivně očekáváme, že body blíže kruhům budou mít větší příslušnost než body vzdálenější. Jenže kvůli prvnímu pravidlu a stejné vzdálenosti od obou kruhů budou mít všechny body přímkou příslušnost k oběma kruhům 0,5, ať jsou blízko či daleko. Obdobným případem je také bod P úplně vlevo na obr 2.1. Evidentně nepatří do žádného kruhu, a tak by měl mít nízký stupeň příslušnosti k oběma. Jelikož se nachází dále od pravého kruhu, bude k němu mít menší příslušnost, řekněme  $\mu(x, k_p) = 0,2$ . Díky normalizaci (první podmínka) však automaticky vzroste jeho příslušnost k levému kruhu  $\mu(x, k_l) = 1 - 0,2 = 0,8$ . Řešení tohoto problému nabízí possibilistická shluková analýza.

## 2.5 Possibilistická fuzzy shluková analýza

Je-li splněna následující podmínka, jedná se o possibilistické dělení (viz.[1]):

$$\forall k \in K : \sum_{x \in X} \mu(x, k) > 0, \quad (2.11)$$

kde  $X$  je množina objektů, které budeme analyzovat, a  $K$  je množina všech shluků. Jak vidíme, possibilistické dělení je zobecněním probabilistického. Platí tedy, že každé probabilistické dělení je zároveň i possibilistické, ne však naopak. To, že zde nemáme podmínku normalizace, je výhodné obzvláště u analýzy dat zatížených šumem. Bodům,

jenž očividně nepatří k žádnému shluku (na obr 2.1 bod P), dáme nízký stupeň příslušnosti k pravému i levému kruhu, například  $\mu(x, k_p) = 0,2$  a  $\mu(x, k_l) = 0,3$ .

## 2.6 Objektivní funkce

Objektivní funkce  $J(\mu)$  se zavádí jako vyhodnocovací kritérium pro změření kvality různých řešení stejného problému. Protože hodnota této funkce vyjadřuje chybu měření, hledáme při shlukové analýze její minimum. Obecně lze objektivní funkci definovat různými způsoby, záleží jen na problému, který řešíme. Pro probabilistickou shlukovou analýzu je vhodné (podle [1]) zavést objektivní funkci ve tvaru:

$$J(\mu) = \sum_{x \in X} \sum_{k \in K} \mu^m(x, k) \cdot d^2(x, k), \quad (2.12)$$

kde  $X$  je množina všech klasifikovaných bodů, a  $K$  množina shluků. Písmenem  $m$  je zde označen takzvaný fuzzyfikační koeficient. Pokud se  $m$  blíží shora k 1, přejde nám fuzzy shluková analýza v klasickou (ostrou) shlukovou analýzu a funkce příslušnosti bude nabývat buď hodnoty 0 nebo 1. Pro  $m$  blížíící se k nekonečnu pak bude mít každý bod příslušnost  $\mu = 1/c$ , kde  $c$  je počet shluků. Čím je tedy  $m$  větší, tím je výsledek více „fuzzy“. Obecně se doporučuje volit koeficient  $m = 2$ . Pokud zavedeme objektivní funkci takto, pak nám minimalizace dá tento vztah pro funkci příslušnosti:

$$\mu(x, k) = \frac{1}{\sum_{j \in K} (d(x, k) / d(x, j))^{\frac{2}{m-1}}} \quad (2.13)$$

Obdobně je zavedena objektivní funkce pro possibilistickou shlukovou analýzu:

$$J(\mu) = \sum_{x \in X} \sum_{k \in K} \mu^m(x, k) \cdot d^2(x, k) + \sum_{k \in K} \eta(k) \sum_{x \in X} (1 - \mu(x, k))^m, \quad (2.14)$$

jejíž minimalizací dostaneme vztah pro funkci příslušnosti:

$$\mu(x, k) = \frac{1}{1 + (d^2(x, k) / \eta(k))^{\frac{1}{m-1}}}. \quad (2.15)$$

Pomocí vhodně navržené funkce  $\eta(k)$  můžeme ovlivňovat tvar a velikost jednotlivých shluků. Například pro  $\eta(k) = d^2(x, k)$ ,  $m = 2$  dostáváme  $\mu(x, k) = 0,5$ . Tímto postupem si můžeme pro každý shluk definovat vzdálenost, v jaké má být funkce příslušnosti rovna 0,5. Nemáme-li však předem žádnou informaci o tvarech shluků či jejich velikostech, můžeme funkci  $\eta(k)$  odhadnout pomocí vzorce:

$$\eta(k) = \frac{\sum_{x \in X} \mu^m(x, k) \cdot d^2(x, k)}{\sum_{x \in X} \mu^m(x, k)} \quad (2.16)$$

## 2.7 Základní algoritmy fuzzy shlukové analýzy

Níže uvedené algoritmy historicky vychází z k-means algoritmu, jenž provádí ostré dělení do shluků (prvek ke shluku patří, anebo ne). K zobecnění pro fuzzy probabilistické shlukování došlo nejprve zavedením objektivní funkce (2.12) s pevným fuzzifikačním koeficientem  $m = 2$ , a poté rozšířením  $m$  na  $m > 1$ ,  $m \in \mathfrak{R}$ . Zájemce o více informací můžeme odkázat na [1], [5].

Nechť máme množinu dat  $X = \{x_1, x_2, \dots, x_n\}$ . A hledáme prvky z množiny shluků  $K = \{k_1, k_2, \dots, k_c\}$ . Matice příslušnosti  $\mathbf{U}$  je typu  $n \times c$ , kde každý prvek  $\mathbf{U}_{i,j} = \mu(x, k)$ .

### 2.7.1 Algoritmus 1, probabilistická SA

1) Inicializace:

- a) Zvol počet shluků  $c$  tak, aby:  $2 < c < n$ ;  $c \in \mathbf{N}$ .
- b) Zvol fuzzifikační koeficient  $m$  tak, aby  $m \in \mathbf{R}$ ,  $m > 1$ .
- c) Zvol ukončovací koeficient  $\varepsilon > 0$ .
- d) Ukazatel kroků  $l$  nastav na  $l = 0$ .
- e) Inicializuj matici příslušnosti  $\mathbf{U}^{(0)}$  náhodnými čísly.

2) OPAKUJ

- a) Zvětši  $l$  o 1.
- b) Urči středy shluků  $K^{(l)}$  tak, aby objektivní funkce  $J$  byla minimální pro  $\mathbf{U}^{(l-1)}$ .
- c) Urči  $\mathbf{U}^{(l)}$  pro středy shluků  $K^{(l)}$  podle vzorce 2.13.

3) DOKUD JE:  $\|\mathbf{U}^{(l)} - \mathbf{U}^{(l-1)}\| > \varepsilon$ .

Ukazuje se, že používat tento algoritmus pro possibilistickou SA není vhodné. Ale je výhodné jej použít pro její inicializaci. Napřed vykonáme algoritmus 1 a

z jím nalezených středů shluků  $k_i$  a z matice příslušnosti  $\mathbf{U}$  určíme hodnoty  $\eta(k)$ . Poté vykonáme possibilistickou SA (tím je myšleno algoritmus 1, kde se v bodě 2.c počítá  $\mathbf{U}$  podle vzorce 2.15) a z výsledků jí poskytnutých ještě jednou odhadneme  $\eta(k)$ . Nakonec provedeme ještě jednou possibilistickou SA. Jak je vidět, algoritmus pro possibilistickou analýzu je časově náročnější než pro probablistickou SA. Pro jeho vykonání prakticky musíme provést třikrát algoritmus 1.

## 2.7.2 Algoritmus 2, possibilistická SA

1) Inicializace:

- a) Zvol počet shluků  $c$  tak, aby:  $2 < c < n$ ;  $c \in \mathbf{N}$ .
- b) Zvol fuzzifikační koeficient  $m$  tak, aby  $m \in \mathbf{R}$ ,  $m > 1$ .
- c) Zvol ukončovací koeficient  $\varepsilon > 0$ .
- d) Ukazatel kroků  $l$  nastav na  $l = 0$ .
- e) Inicializuj matici příslušnosti  $\mathbf{U}^{(0)}$  náhodnými čísly.

2) Vykonej algoritmus pro probablistickou SA.

3) OPAKUJ DVAKRÁT:

- a) Inicializuj  $\mathbf{U}^{(0)}$  a  $K^{(0)}$  předchozími výsledky, nastav ukazatel kroků  $l = 0$ .
- b) Inicializuj  $\eta(k)$  podle vzorce 2.16.
- c) OPAKUJ
  - i) Zvětš  $l$  o 1.
  - ii) Urči středy shluků  $K^{(l)}$  tak, aby  $J$  byla minimální pro  $\mathbf{U}^{(l-1)}$ .
  - iii) Urči  $\mathbf{U}^{(l)}$  pro středy shluků  $K^{(l)}$  podle vzorce 2.15.
- d) DOKUD JE:  $\|\mathbf{U}^{(l)} - \mathbf{U}^{(l-1)}\| > \varepsilon$ .

4) KONEC „OPAKUJ DVAKRÁT“.

## 2.8 Fuzzy c-means algoritmus (FCM), probabilistická verze

Tento algoritmus vychází z algoritmu 1. Slouží k nalezení předem známého počtu shluků přibližně stejné velikosti. Každý shluk má tvar kruhu (v případě vícedimenzionálních dat hyperkoule) a je charakterizován svým středem. Nalézají-li se v blízkosti shluku jiné shluky, je jeho tvar deformován. Pro výpočet vzdálenosti je použita Eukleidovská vzdálenost. Výhodou FCM je rychlost výpočtu. Jelikož je algoritmus iterační, nejsme schopni předem říci, jak dlouho bude algoritmus trvat, ani po kolika krocích bude splněna ukončovací podmínka. Toto může být nevýhodné obzvláště při real-time aplikacích. V některých případech lze rychlost FCM zvětšit. Například pokud sledujeme pohybující se objekt, můžeme každé spuštění algoritmu inicializovat ne náhodně, ale minulým výsledkem. Tak dojde k splnění ukončovací podmínky dříve (po méně iteracích).

### Popis FCM algoritmu:

#### 1. Inicializace:

- a) Zvol počet shluků  $c$  tak, aby:  $2 < c < n$ ;  $c \in \mathbf{N}$ .
- b) Zvol koeficient  $m$  tak, aby  $m > 1$ .
- c) Zvol ukončovací koeficient  $\varepsilon > 0$ .
- d) Ukazatel kroků  $l$  nastav na  $l = 0$ .
- e) Inicializuj matici příslušnosti  $\mathbf{U}^{(0)}$  náhodnými čísly.

#### 2. OPAKUJ:

- a) Zvětši  $l$  o 1.
- b) Vypočti středy shluků  $k_i$  podle vzorce:

$$k_i^{(l)} = \frac{\sum_{k=1}^N (\mu^{(l-1)}(k, i))^m x_k}{\sum_{k=1}^N (\mu^{(l-1)}(k, i))^m}, \quad 1 \leq i \leq c. \quad (2.17)$$

#### 3) Vypočti vzdálenost jednotlivých bodů ke středům shluků $k_i$ :

$$d^2(k, i) = (x_k - k_i)^T (x_k - k_i), \quad 1 \leq i \leq c, \quad 1 \leq k \leq N. \quad (2.18)$$

#### 4) Vypočti novou matici příslušnosti s nově přepočítanými středy shluků:

$$\mu^{(l)}(k, i) = \frac{1}{\sum_{j=1}^c (d(k, i)/d(k, j))^{2/m-1}}, \quad 1 \leq i \leq c, \quad 1 \leq k \leq N. \quad (2.19)$$



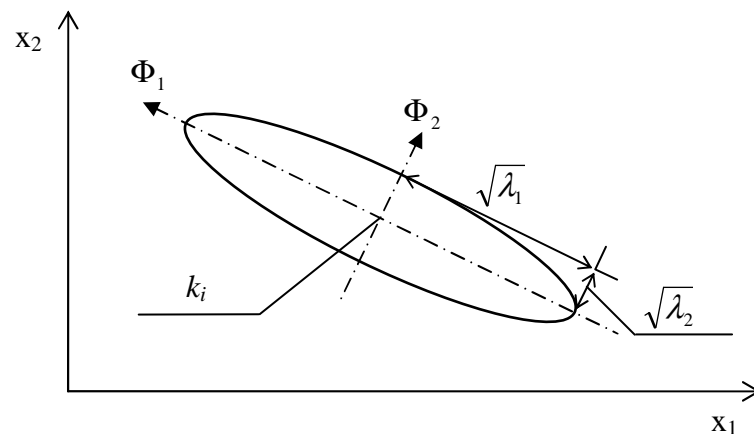
5) DOKUD JE:  $\|\mathbf{U}^{(l)} - \mathbf{U}^{(l-1)}\| > \varepsilon$ .

## 2.9 Gustafson-Kesselův algoritmus (GK), probabilistická verze

GK také vychází z algoritmu 1. Nepracuje se zde s Eukleidovskou vzdáleností jako u FCM, nýbrž s Mahalanobisovou vzdáleností (2.8). Pro výpočet kovarianční matice je použit vztah:

$$\mathbf{A}_i = \eta \sqrt{\det(\mathbf{S}_i)} \mathbf{S}_i^{-1}. \quad (2.20)$$

$\mathbf{S}_i$  je fuzzy kovarianční matice a vzorec pro její výpočet je uveden v 2.23. Díky použití této metriky je GK schopen nalézt shluky ve tvaru různě natočených elips (v případě vícedimenzionálních dat elipsoidů). Tvar každé elipsy je dán vlastními čísly  $\lambda_i$  a její natočení vlastními vektory  $\Phi_i$  příslušné fuzzy kovarianční matice  $\mathbf{S}_i$  [2].



Obr. 2.2 Natočení elipsoidu vyhledaného GK algoritmem

Všechny elipsy mají stejný obsah, což vyplývá z podmínky stejně velkých determinantů kovariančních matic:

$$\det(\mathbf{A}_i) = 1. \quad (2.21)$$

Kvůli použití Mahalanobisovy vzdálenosti je GK výpočetně náročnější, a tudíž pomalejší než FCM. V každém iteračním kroku se musí vypočítat  $c$  fuzzy kovariančních matic  $\mathbf{S}_i$ , jejich determinanty a inverzní matice, aby se mohly spočítat kovarianční matice  $\mathbf{A}_i$ .

## Popis Gustafson-Kesselova algoritmu:

### 1. Inicializace:

- Zvol počet shluků  $c$  tak, aby:  $2 < c < n$ ;  $c \in \mathbf{N}$ .
- Zvol koeficient  $m$  tak, aby  $m > 1$ .
- Zvol ukončovací koeficient  $\varepsilon > 0$ .
- Ukazatel kroků  $l$  nastav na  $l = 0$ ,
- Inicializuj matici příslušnosti  $\mathbf{U}^{(0)}$  náhodnými čísly.

### 2. OPAKUJ:

- Zvětš  $l$  o 1.
- Vypočti středy shluků  $k_i$  podle vzorce:

$$k_i^{(l)} = \frac{\sum_{k=1}^N (\mu^{(l-1)}(k, i))^m x_k}{\sum_{k=1}^N (\mu^{(l-1)}(k, i))^m}, \quad 1 \leq i \leq c. \quad (2.22)$$

### 3) Pro každý shluk spočti jeho fuzzy kovarianční matici $\mathbf{S}_i$ :

$$\mathbf{S}_i^{(l)} = \frac{\sum_{k=1}^N (\mu^{(l-1)}(k, i))^m (x_k - k_i)(x_k - k_i)^T}{\sum_{k=1}^N (\mu^{(l-1)}(k, i))^m}, \quad 1 \leq i \leq c. \quad (2.23)$$

### 4) Vypočti vzdálenost jednotlivých bodů ke středům shluků $k_i$ :

$$d^2(k, i) = (x_k - k_i^{(i)})^T \left[ \sqrt{\det(\mathbf{S}_i)} \mathbf{S}_i^{-1} \right] (x_k - k_i^{(i)}), \quad 1 \leq i \leq c, 1 \leq k \leq N. \quad (2.24)$$

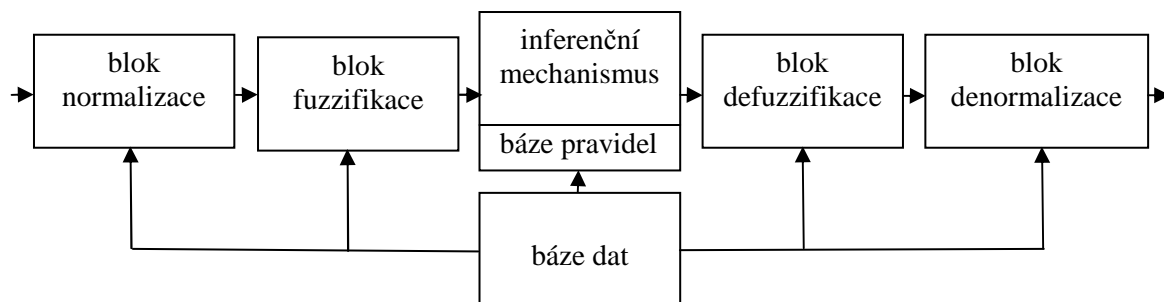
### 4) Vypočti novou matici příslušnosti:

$$\mu^{(l)}(k, i) = \frac{1}{\sum_{j=1}^c (d(k, i)/d(k, j))^{m-1}}, \quad 1 \leq i \leq c, 1 \leq k \leq N. \quad (2.25)$$

### 5) DOKUD JE: $\|\mathbf{U}^{(l)} - \mathbf{U}^{(l-1)}\| > \varepsilon$ .

### 3 Fuzzy systémy

Fuzzy systémy rozumíme takové systémy, které pracují s fuzzy množinami. Můžou mít fuzzy proměnné na svém vstupu či výstupu, anebo s nimi pracují jen uvnitř a navenek se tváří jako klasické systémy. Struktura takového systému je znázorněna na obrázku 3.1.

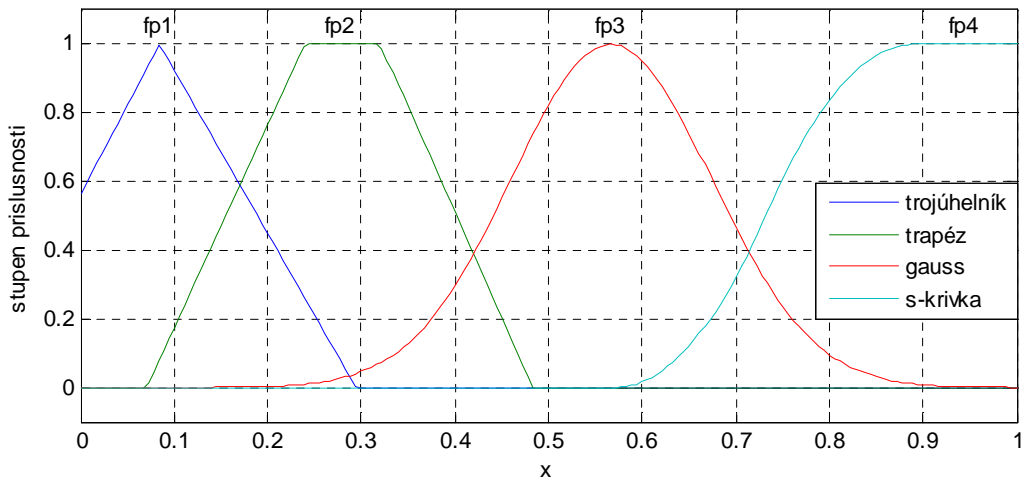


Obr. 3.1 Fuzzy systém

Podíváme-li se podrobněji na schéma fuzzy systému, vidíme, že je „symetrický“. Vstupnímu bloku normalizace odpovídá výstupní blok denormalizace a bloku fuzzifikace blok defuzzifikace. Nyní si probereme podrobněji jednotlivé bloky. Každá vstupní veličina může nabývat hodnot v určitém rozsahu. Tomu se říká, že je definovaná na nějakém universu. Různé vstupní veličiny mohou mít různá universa. Je výhodné pracovat s jednotným, normalizovaným universem, například interval  $\langle 0,+1 \rangle$ . Proto se v bloku normalizace provádí transformace vstupních veličin tak, aby po jejím provedení měly všechny vstupní veličiny stejné universum.

V bloku fuzzifikace dochází k převedení ostré hodnoty vstupní veličiny na příslušnost k několika fuzzy množinám (termům), jenž bezzbytku pokrývají normalizované universum. Termy jsou reprezentovány svými funkcemi příslušnosti. Ty mohou mít různý tvar, například trojúhelníková funkce, lichoběžníková (trapézová) funkce, Gaussova funkce nebo Zadehova  $S$ -funkce. Pokrytí universa  $\langle 0,+1 \rangle$  některými běžně používanými funkcemi příslušnosti je ukázáno na obrázku 3.2. Například bude-li mít vstupní veličina  $x$  ostrou hodnotu 0,4, pak bude patřit k termu  $fp2$  s příslušností  $\mu_{x,fp2} = 0,5$  a k termu  $fp3$  s příslušností  $\mu_{x,fp3} = 0,3$ . K termům  $fp1$  a  $fp4$  nebude

náležet vůbec:  $\mu_{x,fp1} = \mu_{x,fp2} = 0$ . Obdobně, jako je rozděleno vstupní universum  $x$ , je také rozděleno mezi několik fuzzy množin výstupní universum  $w$ .



Obr. 3.2 Funkce příslušnosti jednotlivých fuzzy množin charakterizujících vstupní veličinu  $x$ .

Jádrem fuzzy systému je jednotka obsahující inferenční mechanismus a bázi pravidel. Pravidla jsou ve formě fuzzy implikace a skládají se z podmínkové části (antecedent) a části závěrové (konsekvent). Podmínková část je tvořena fuzzy výrokem, to jest několika termy pospojovanými logickými spojkami (AND, OR). V případě dvou vstupů obsahuje antecedent dva termy, v případě tří vstupů tři termy a podobně. Závěrová část je tvořena buď fuzzy výrokem (v případě Mamdani fuzzy systému), anebo matematickými funkcemi (Takagi-Sugeno systémy).

Jestliže je  $k$ -tá fuzzy množina příslušného vstupu  $x$  nebo  $y$  označena  $termX.k$ ,  $termY.k$ , výstupní fuzzy množina  $termV.k$ , pak část báze pravidel může vypadat takto:

### Mamdani fuzzy systémy

$$\text{IF } (x \text{ IS } termX.1 \text{ AND } y \text{ IS } termY.3) \text{ THEN } w \text{ IS } termV.1 \quad (3.1)$$

$$\text{IF } (x \text{ IS } termX.3 \text{ AND } y \text{ IS } termY.4) \text{ THEN } w \text{ IS } termV.3$$

### Takagi-Sugeno fuzzy systémy

$$\text{IF } (x \text{ IS } termX.1 \text{ AND } y \text{ IS } termY.3) \text{ THEN } w = f_1(x,y) \quad (3.2)$$

$$\text{IF } (x \text{ IS } termX.3 \text{ AND } y \text{ IS } termY.4) \text{ THEN } w = f_2(x,y)$$

Funkce  $f_i$  mohou být libovolné, zpravidla jsou lineární, takže popisují přímky, roviny nebo nadroviny.

### 3.1 Fuzzy logické operace

V případě, že by podmínková i závěrová část pravidla byly ostré výroky, pak by NOT, AND, OR, IF-THEN byly klasické logické operace. Protože jsou však namísto ostrých výroků použity fuzzy výroky, musí být použity i fuzzy negace, konjunkce, disjunkce a implikace.

Unární operace NOT (negace), binární operace AND (konjunkce) a OR (disjunkce) mohou být ve fuzzy logice definovány několika způsoby. Pro dvě proměnné  $\alpha, \beta \in \langle 0,1 \rangle$  uvedeme pro každou operaci dvě z mnoha možných variant definic, jenž jsou uvedeny v [3]:

$$\text{Standardní negace: } \text{NOT } \alpha = 1 - \alpha. \quad (3.3)$$

$$\text{Yagerova negace: } \text{NOT } \alpha = \sqrt[w]{1 - \alpha^w}, \quad w > 0. \quad (3.4)$$

$$\text{Standardní konjunkce: } \alpha \text{ AND } \beta = \min(\alpha, \beta). \quad (3.5)$$

$$\text{Součinnová konjunkce: } \alpha \text{ AND } \beta = \alpha \beta. \quad (3.6)$$

$$\text{Standardní disjunkce: } \alpha \text{ OR } \beta = \max(\alpha, \beta). \quad (3.7)$$

$$\text{Součinnová disjunkce: } \alpha \text{ OR } \beta = \alpha + \beta - \alpha \beta. \quad (3.8)$$

Můžeme si všimnout, že pokud dosadíme za  $\alpha, \beta \in \{0,1\}$ , mají fuzzy operace stejný výsledek jako klasické ostré operace. Této vlastnosti se říká, že fuzzy operace splňují okrajové podmínky. Můžeme tedy hovořit o zobecnění ostrých operací. Při definici fuzzy implikace se vychází z ostré implikace. Přitom se požaduje, aby fuzzy implikace splňovala okrajové podmínky. Klasickou implikaci lze pomocí logických operací NOT, AND a OR vyjádřit dvěma způsoby:

$$\text{IF } \alpha \text{ THEN } \beta = \text{NOT } \alpha \text{ AND } \beta, \quad (3.9)$$

$$\text{IF } \alpha \text{ THEN } \beta = (\alpha \text{ AND } \beta) \text{ OR NOT } \alpha. \quad (3.10)$$

Každá z těchto definic vede ve fuzzy logice na úplně odlišnou třídu implikací. Uvědomíme-li si navíc, že pro každou ostrou logickou operaci máme několik fuzzy operací, zjistíme, že můžeme vybírat z velkého počtu fuzzy implikací. Přesto se v regulační praxi používá nejčastěji Mamdani implikace, která však okrajové podmínky nespĺňuje a v pravém slova smyslu se ani o implikaci nejedná:

$$\text{IF } \alpha \text{ THEN } \beta = \min(\alpha, \beta). \quad (3.11)$$

### 3.2 Vyvozování a defuzzifikace v Mamdani systémech

Podíváme-li se ještě jednou na strukturu pravidla (3.1), zjistíme, že v jeho podmínkové části jsou dvě ostré hodnoty vstupních veličin. Těm je v bloku fuzzifikace přidělena příslušnost k jednotlivým fuzzy množinám ( $termX.k$ ,  $termY.k$ ). Nad těmito příslušnostmi je poté provedena fuzzy konjunkce (AND), a tak je pro každé pravidlo  $i$  získána hodnota antecedentu  $\mu_{A_i}$ . Z této hodnoty a z příslušné výstupní fuzzy množiny vyskytující se v závěrové části pravidla ( $termV.k$ ) je poté vyvozena nová funkce příslušnosti  $\mu_{K_i}(w)$ . Výsledkem vykonání pravidla tedy není pouze jedna ostrá hodnota, nýbrž fuzzy množina. Všechny fuzzy množiny  $\mu_{K_i}(w)$  jsou sjednoceny pomocí logických spojek OR do jedné fuzzy množiny reprezentované funkcí příslušnosti  $\mu_K(w)$ . V případě defuzzifikační metody centroidů [2] je pak jako ostrá výstupní hodnota  $w_T$  brána souřadnice těžiště plochy tvořené funkcí příslušnosti  $\mu_K(w)$ . Souřadnice těžiště je počítána obdobně jako statistická střední hodnota:

$$w_T = \frac{\int_{-\infty}^{+\infty} w \mu_K(w) dw}{\int_{-\infty}^{+\infty} \mu_K(w) dw}. \quad (3.12)$$

### 3.3 Vyvozování a defuzzifikace v Takagi-Sugeno systémech

Jak již bylo zmíněno, pravidla Takagi-Sugeno systému (3.2) se od pravidel Mamdani systému (3.1) liší pouze v závěrové části. Tento rozdíl znamená jak výraznou změnu v samotném vyvozování, tak i v metodě defuzzifikace. Podobně jako v Mamdani systémech je pro každé pravidlo  $i$  zjištěna hodnota antecedentu  $\mu_{A_i}$ . Ostrá výstupní hodnota  $w_T$  se počítá jako vážený průměr jednotlivých výstupů  $w_i$ , kde vážíme pomocí hodnoty příslušného antecedentu  $\mu_{A_i}$ :

$$w_T = \frac{\sum_{i=1}^m \mu_{A_i} w_i}{\sum_{i=1}^m \mu_{A_i}}. \quad (3.13)$$

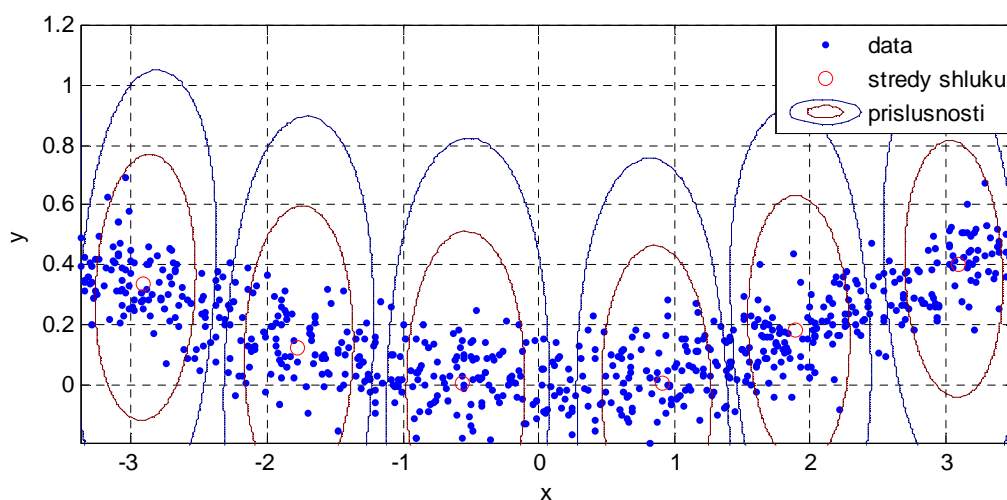
## 4 Aproximace funkcí

V této části se zaměříme na aproximaci funkcí. K identifikaci naměřených dat použijeme fuzzy shlukovou analýzu (kap. 2) a pomocí jejích výsledků nastavíme parametry fuzzy systému (kap. 3) tak, aby co nejlépe approximoval funkční závislost v naměřených datech.

### 4.1 Aproximace funkcí pomocí algoritmu FCM

V předchozích kapitolách jsme si vysvětlili, jak algoritmus FCM pracuje. Nyní si probereme jeho konkrétní využití pro aproximaci funkcí. Naším úkolem tedy bude najít pro množinu naměřených bodů funkci, která popisuje vztah mezi nimi. Při výkladu se zaměříme na data z dvojdimenzionálního prostoru  $\mathbf{R}^2$ , budeme tedy aproximovat funkce jedné proměnné s tím, že obdobný postup je použit i pro vícedimenzionální data.

Použijeme-li algoritmus FCM na množinu dat, vyhledá v nich shluky. Každému datu je přidělena příslušnost ke každému shluku. Jednotlivé shluky jsou charakterizovány pomocí svých středů. Na obrázku 4.1 můžeme vidět výsledek algoritmu. Modře jsou znázorněna data, červeně pak středy shluků. Pomocí elips jsou vyznačeny body, které mají stejnou příslušnost k danému shluku. Body na modrých elipsách mají příslušnost  $\mu(x, k_i) = 0,5$ , body na hnědých elipsách  $\mu(x, k_i) = 0,7$ . Jak si můžeme všimnout, tyto body nepatří mezi klasifikovaná data. Hodnota jejich funkce příslušnosti byla spočtena až po vykonání algoritmu FCM.



Obr. 4.1 FCM: Klasifikace dat do shluků s vyznačenými hodnotami příslušností

modré elipsy:  $\mu(x, k_i) = 0,5$ , hnědé elipsy  $\mu(x, k_i) = 0,7$ .

Dříve jsme se zmiňovali o tom, že výsledkem FCM by měly být kruhové shluky. Na obrázku však vidíme shluky elipsovitého tvaru. To je důsledkem použití probabilistické verze FCM, kdy hodnota  $\mu(x, k_i)$  nezávisí pouze na vzdálenosti k vlastníku shluku, ale je ovlivněna i vzdálenostmi ke shlukům ostatním (2.13). Například pro levý prostřední shluk vidíme, že ve svislém směru klesá příslušnost mnohem pomaleji než ve směru horizontálním, kde je ovlivněna sousedními shluky. Použili-li bychom possibilistický FCM (2.15), k deformaci kružnic na elipsy by nedošlo.

## 4.2 Nastavení parametrů Mamdani systému

Pro správnou funkci každého fuzzy systému je nejprve nutné sestavit bázi pravidel, podle které probíhá inferenční mechanismus. Každé pravidlo reprezentuje jeden shluk. Počet vstupů systému (a tedy i počet termů v antecedentu pravidla) je dán počtem nezávislých proměnných. Dále je potřeba zvolit typ funkcí příslušnosti, které jsou použity v podmínkové a v závěrové části pravidla. Jako nejpřirozenější se zdá zvolit Gaussovu křivku (obr. 3.2, křivka fp3), jenž je popsána rovnicí:

$$g(x) = e^{-\frac{(x-k)^2}{2\sigma^2}} \quad (4.1)$$

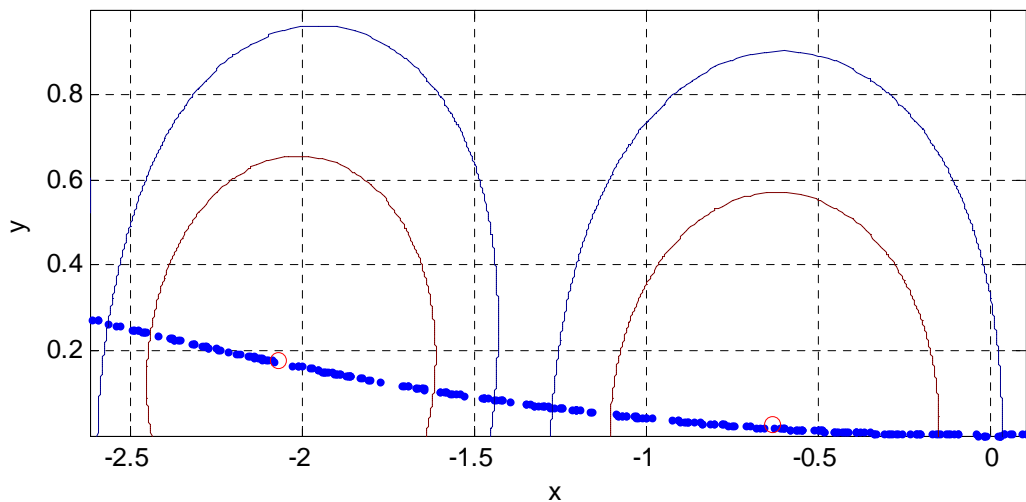
Pro každou funkci příslušnosti je tedy nutné zjistit dva parametry, posun  $k$  a směrodatnou odchylku  $\sigma$  (tak je ve statistice běžně označován tento koeficient Gaussovy křivky, i když v našem případě se o statistickou směrodatnou odchylku nejedná). Pro hodnotu  $k$  můžeme použít středy nalezených shluků  $k_i$ . Nyní potřebujeme určit parametr  $\sigma$  tak, aby funkce  $g(x)$  co nejvíce odpovídala průmětu funkce příslušnosti  $\mu(x, k_i)$  příslušného shluku  $k_i$  do dané osy. Protože se jedná o obecně odlišné funkce zvonovitého tvaru, stanovili jsme si, že se obě funkce budou shodovat v bodě:

$$g(x) = e^{-0,5} \cong 0,606. \quad (4.2)$$

Tento bod odpovídá vzdálenosti od vrcholu křivky  $x - k = \sigma$ . Podíváme-li se na obrázek 4.2, vidíme, že průmět shluků do osy  $x$  celkem odpovídá rozmezí, ve kterém funkce prochází přes shluk (průmětem budeme dále rozumět průmět bodů na elipse,



majících příslušnost 0,606). Naopak průmět shluku do osy  $y$  je mnohem širší než přírůstek funkce v rozmezí shluku. Použili-li bychom pro určení směrodatné odchylky průmět shluku do osy  $y$ , získali bychom směrodatnou odchylku přibližně  $\sigma = 0,6$ . Přírůstek naší funkce v rozmezí shluku je však jen 0,2, což odpovídá směrodatné odchylce  $\sigma = 0,1$ . S ohledem na průběh funkce jsme se rozhodli celou věc pojmout z trochu jiné stránky. Pro určení parametru  $\sigma$  nepoužijeme průmět celého shluku do jednotlivých os, ale vezmeme v úvahu pouze předložená data. Z těch vybereme ty, které mají příslušnost 0,6. Jako parametr  $\sigma$  pak vezmeme pro každou osu absolutní hodnotu rozdílu příslušných souřadnic bodu a středu  $k$ . Aby byla metoda odolnější, vybereme body z rozmezí příslušností 0,5 až 0,7 (Na obrázku modrá a hnědá křivka). Tak pro každý bod  $i$ , jenž se nachází v tomto pásmu, dostaneme příslušný koeficient  $\sigma_i$ . Výsledný koeficient  $\sigma$  je pak brán jako aritmetický průměr jednotlivých  $\sigma_i$ . Tato metoda je navržena pro data, která nejsou zatížena šumem. Čím více budou data rozmístěna po celém „mezikruží“, tím bude odhad  $\sigma$  nepřesnější. K určitému zlepšení výsledků funkce také přispívá fakt, že bylo použito průměrování z více bodů, které citlivost metody na šum zmenší.

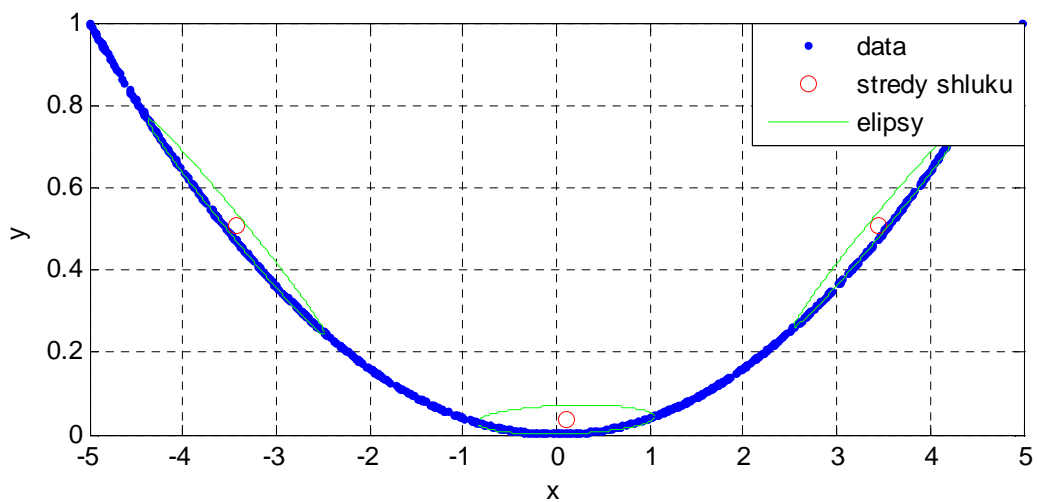


Obr. 4.2 FCM: 2 shluky pro nezašumělá data

### 4.3 Aproximace funkcí pomocí GK

Obdobně jako v předchozí části i zde se omezíme na výklad aproximace funkcí jedné proměnné s tím, že postup lze analogicky rozšířit pro vícerozměrné funkce. Pro

zpracování výsledků algoritmu GK pro účely aproximace je jako stvořen Takagi-Sugeno fuzzy systém. Konsekventní části jeho pravidel (3.2) jsou totiž tvořeny matematicky zadanými funkcemi, v našem případě rovnicemi přímek (pro vícedimenzionální data rovin a nadrovin). Naším úkolem tedy bude zpracovat výsledky Gustafson-Kesselova algoritmu tak, abychom získali koeficienty těchto přímek. Jak už víme z minulých kapitol, výstupem GK jsou shluky ve tvaru elips (viz. obrázky 2.2 a 4.3), jejichž parametry jsou dány fuzzy kovarianční maticí  $\mathbf{S}$  spočtenou podle vzorce 2.23. Principiálně lze získat informace o poloze a natočení elips také z kovarianční matice  $\mathbf{A}$ , ale její použití se nám v praxi neosvědčilo.



Obr. 4.3 GK: Elipsy dané fuzzy kovariančními maticemi  $\mathbf{S}_i$ .

#### 4.4 Nastavení parametrů Takagi-Sugeno systému

Pro nastavení báze pravidel Takagi-Sugeno systému potřebujeme zjistit dva typy parametrů. Pro podmínkové části pravidel je potřeba nastavit konstanty jednotlivých funkcí příslušnosti, a pro závěrové části pravidel musí být zjištěny z náklonu a velikosti shluků koeficienty lineárních funkcí.

Pro fuzzy množiny tvořící antecedent opět zvolíme funkci příslušnosti danou Gaussovou křivkou 4.1. Posun křivky po ose  $x$  je dán středem nalezeného shluku  $k_i$ , šířka křivky, vyjádřena pomocí směrodatné odchylky  $\sigma$ , je pak vypočtena z průmětu elipsy (dané fuzzy kovarianční maticí  $\mathbf{S}$ ) do osy  $x$ . Nyní se pokusíme zodpovědět, jak takový průmět realizovat, máme-li zadánu symetrickou, pozitivně definitní kovarianční matici. Rovnice elipsy posunutá do středu shluku  $k_i$  je dána vztahem:

$$(\mathbf{x} - \mathbf{k})^T \mathbf{S}^{-1} (\mathbf{x} - \mathbf{k}) = 1, \quad (4.3)$$

kde  $\mathbf{x}$  je vektor souřadnic každého bodu ležícího na elipse a  $\mathbf{k}$  je vektor souřadnic středu elipsy. Sklon poloos elipsy je určen vlastními vektory matice  $\mathbf{S}$ . Velikost poloos se vypočte jako druhá odmocnina z příslušných vlastních čísel. Průmět elipsy do osy  $x$  získáme následujícím postupem. Nejprve provedeme normalizaci vlastních vektorů matice  $\mathbf{S}$  na velikost jedna. Poté vynásobíme každý normovaný vlastní vektor  $\Phi_\lambda$  odmocninou z jeho vlastního čísla  $\lambda$ . Jako směrodatnou odchylku  $\sigma_x$  vezmeme největší z  $x$ -ových souřadnic takto získaných vektorů. Když bude mít aproximovaná funkce více nezávislých proměnných, bude postup stejný, jenom výběr maxima, a tedy nalezení šířky funkce příslušnosti, se provede pro každou osu.

Získání lineárních rovnic pro závěrovou část si vysvětlíme na příkladu aproximace funkce dvou proměnných. Výsledkem GK algoritmu jsou v tomto případě shluky představované třídimenziálními elipsoidy. Ty se budeme snažit nahradit rovinou určenou dvěma největšími poloosami elipsoidu. Jinak řečeno, dvěma největšími vlastními vektory fuzzy kovarianční matice  $\mathbf{S}$  posunutými do středu shluku. Víme, že každá rovina je dána bodem, který na ní leží, a normálovým vektorem. Ten je k rovině kolmý. Také víme, že všechny poloosy elipsy, a tedy i její vlastní vektory, jsou na sebe kolmé. Jako normálový vektor vybereme ten vlastní vektor, který přísluší nejmenšímu vlastnímu číslu. Na velikosti tohoto vektoru nezáleží, důležitý je jen jeho směr. Jako bod náležící rovině přirozeně zvolíme střed shluku. Rovnice roviny, ze které budeme vycházet při výpočtu je:

$$ak_1 + bk_2 + ck_3 + d = 0 \quad (4.4)$$

Přitom  $\Phi = (a, b, c)$  je normálový vektor. Pro výpočet koeficientu  $d$  stačí dosadit do rovnice střed shluku  $\mathbf{K} = (k_1, k_2, k_3)$ . Tento postup má tu výhodu, že je velmi snadno rozšiřitelný i na vícedimenziální data, kde má rovnice nadroviny analogický tvar.

## 4.5 Měření kvality aproximace

Aproximaci funkce provedeme následujícím způsobem. Nejprve vygenerujeme podle známé funkce množinu bodů. Na této množině se bude náš algoritmus „učit“. Pomocí FCM nebo GK vyhledá v množině shluky dat a z těchto shluků poté vytvoří fuzzy systém (Mamdani nebo Takagi-Sugeno) reprezentující hledanou funkci. Poté

ověříme, nakolik odpovídá funkce generovaná naším fuzzy systémem funkci hledané. Abychom mohli porovnávat výsledky aproximace různých funkcí generovaných oběma fuzzy systémy pro rozdílný počet shluků, zavedeme pro srovnání různých výsledků směrodatnou odchylku RMSE:

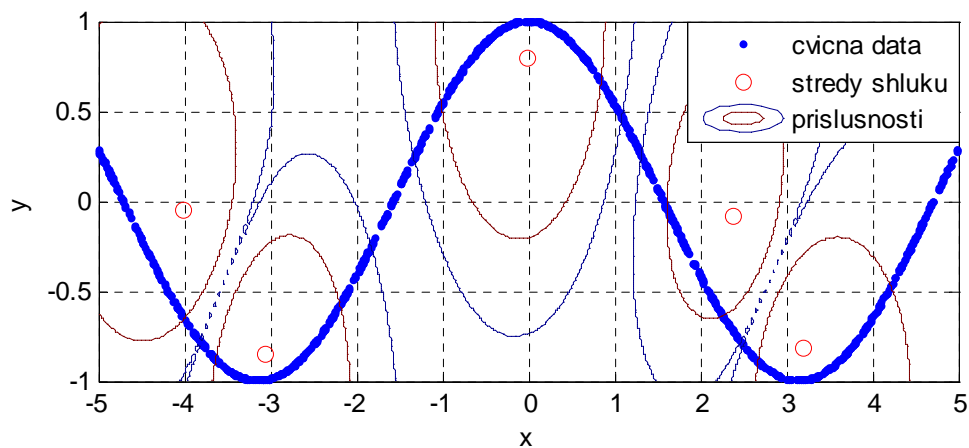
$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x}_i)^2}, \quad (4.5)$$

kde  $x_i - \bar{x}_i$  je rozdíl mezi naším algoritmem generovanou hodnotou  $x_i$  a hodnotou  $\bar{x}_i$ , kterou by měl správně generovat. Dalším parametrem, který nás zajímá, je čas, který daný algoritmus (FCM, GK) potřebuje pro nalezení shluků.

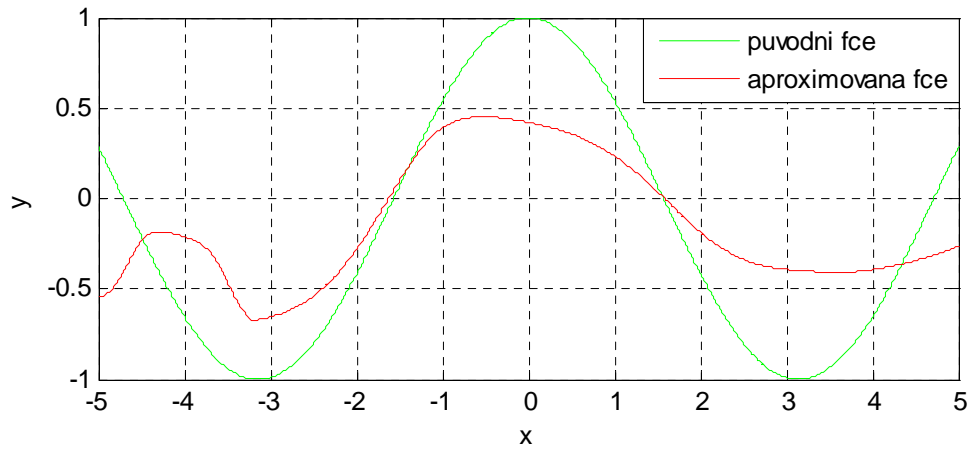
## 4.6 Aproximace funkcí jedné proměnné

### 4.6.1 Aproximace funkce kosinus – FCM

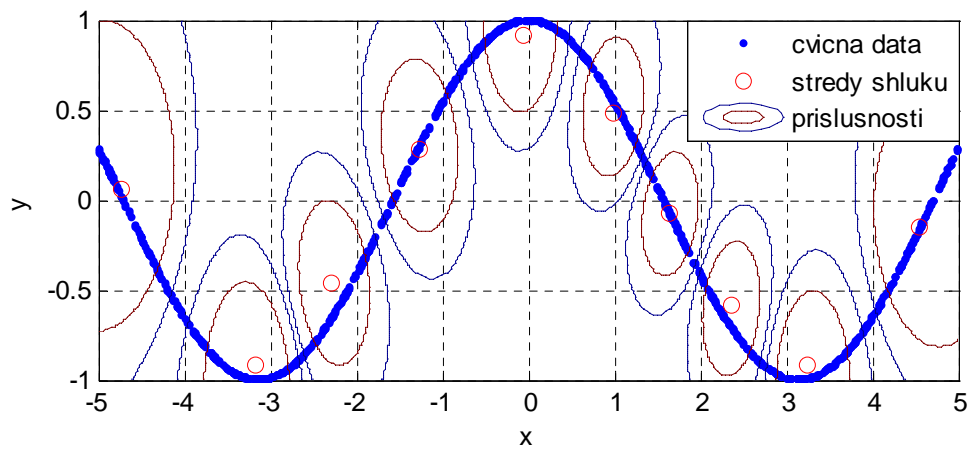
Nejprve budeme aproximovat funkci  $y = \cos x$ , a to pomocí algoritmu FCM spolu s Mamdani fuzzy systémem. Měření provedeme pro různý počet shluků, přičemž výsledky jednotlivých měření budou uvedeny pod jednotlivými obrázky. Pro oba algoritmy byla zvolena množina obsahující 1000 cvičných dat.



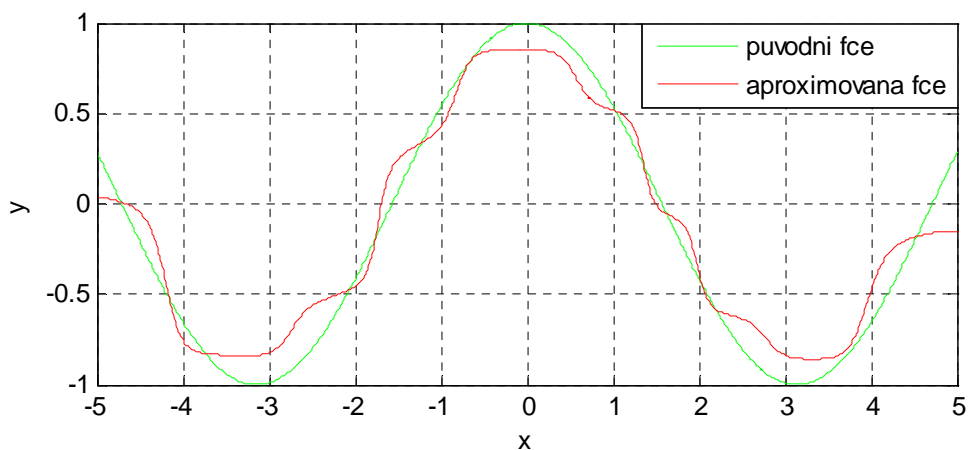
Obr. 4.4.a FCM 5 shluků, cvičná data a nalezené shluky



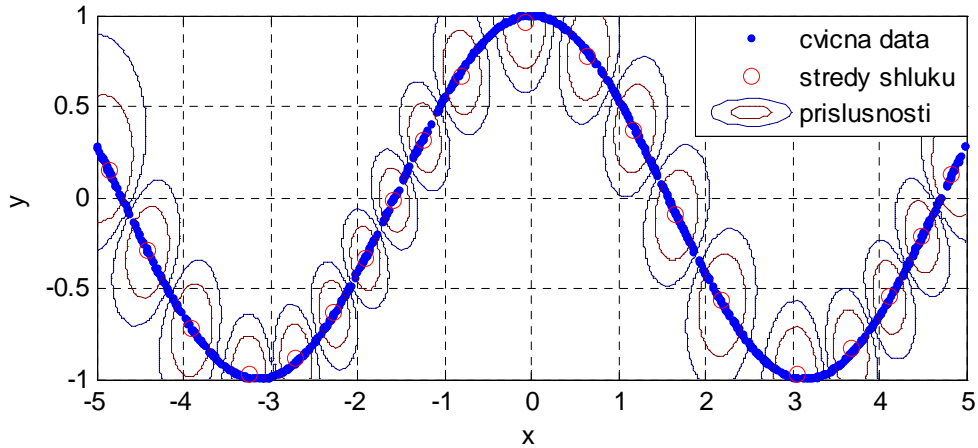
Obr. 4.4.b FCM 5 shluků, výsledek aproximace:  $t = 0,078s$ ,  $RMSE = 0,3816$



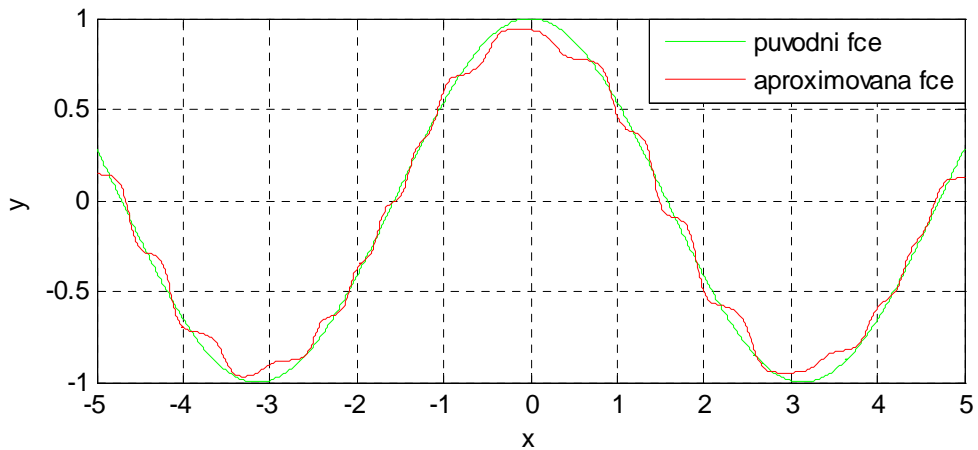
Obr. 4.5.a FCM 10 shluků, cvičná data a nalezené shluky



Obr. 4.5.b FCM 10 shluků, výsledek aproximace:  $t = 0,1410s$ ,  $RMSE = 0,1358$



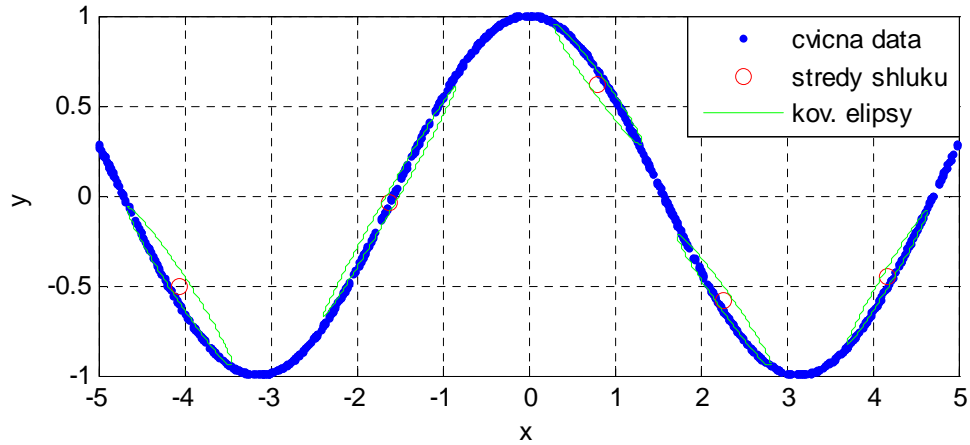
Obr. 4.6.a FCM 20 shluků, cvičná data a nalezené shluky



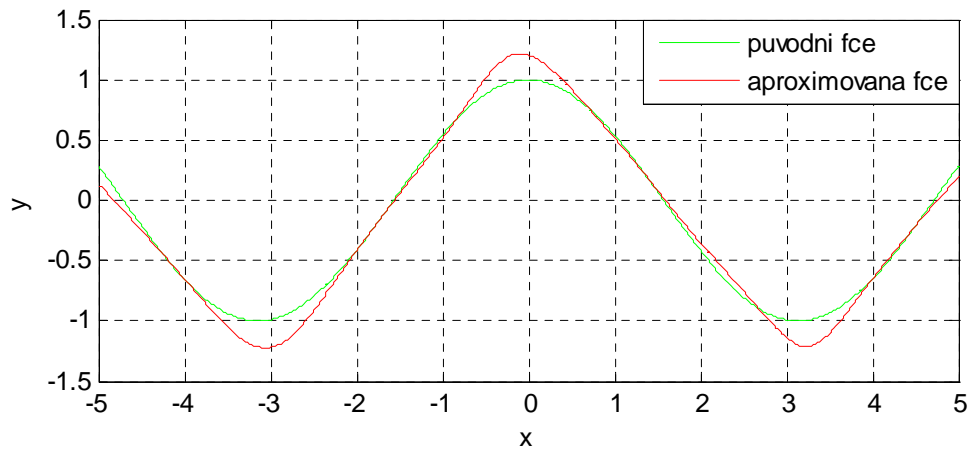
Obr. 4.6.b FCM 20 shluků, výsledek aproximace:  $t = 2,0310s$ ,  $RMSE = 0,0590$

#### 4.6.2 Aproximace funkce kosinus – GK

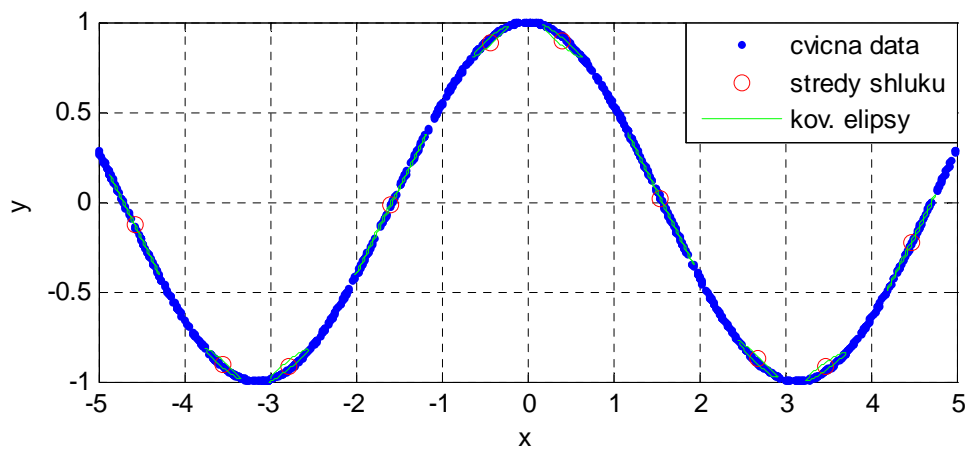
V následující části otestujeme výsledky aproximace pomocí GK spolu s Takagi-Sugeno fuzzy systémem. Měření bude provedeno pro stejný počet shluků jako při testování FCM algoritmu. Na horním obrázku vidíme výsledek GK algoritmu s červeně vyznačenými středy shluků a zelenými elipsami danými fuzzy kovariančními maticemi.



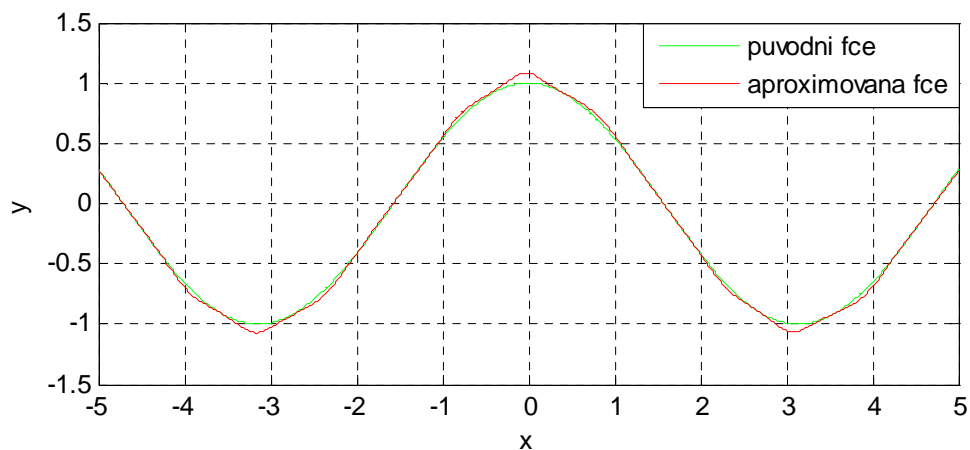
Obr. 4.7.a GK 5 shluků, cvičná data a nalezené shluky



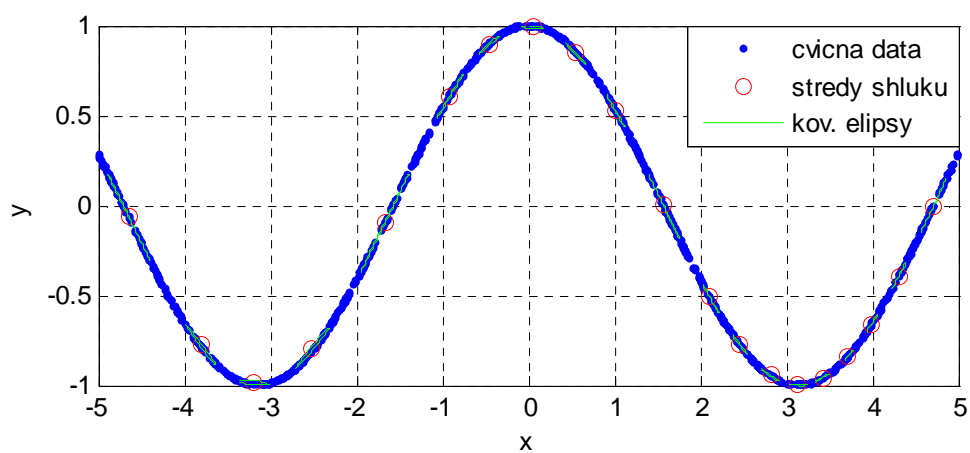
Obr. 4.7.b GK 5 shluků, výsledek aproximace:  $t = 0,1880s$ ,  $RMSE = 0,1031$



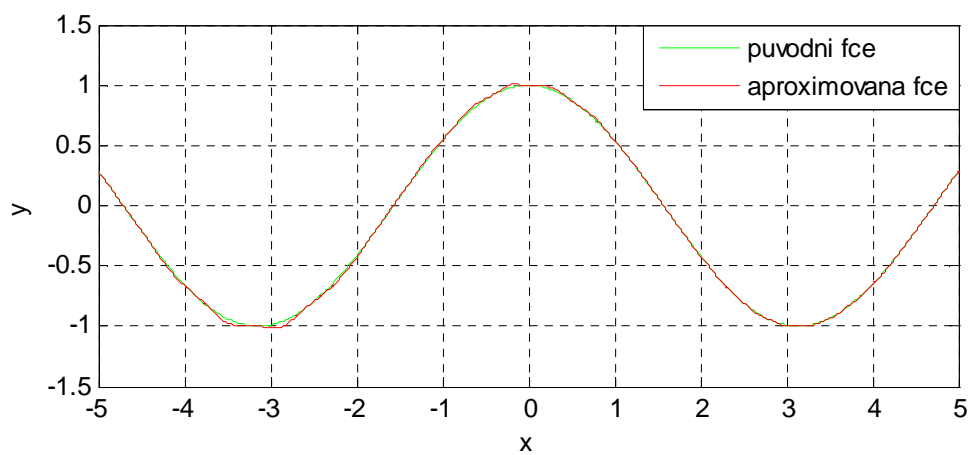
Obr. 4.8.a GK 10 shluků, cvičná data a nalezené shluky



Obr. 4.8.b GK 10 shluků, výsledek aproximace:  $t = 0,6250s$ ,  $RMSE = 0,0275$



Obr. 4.9.a GK 20 shluků, cvičná data a nalezené shluky

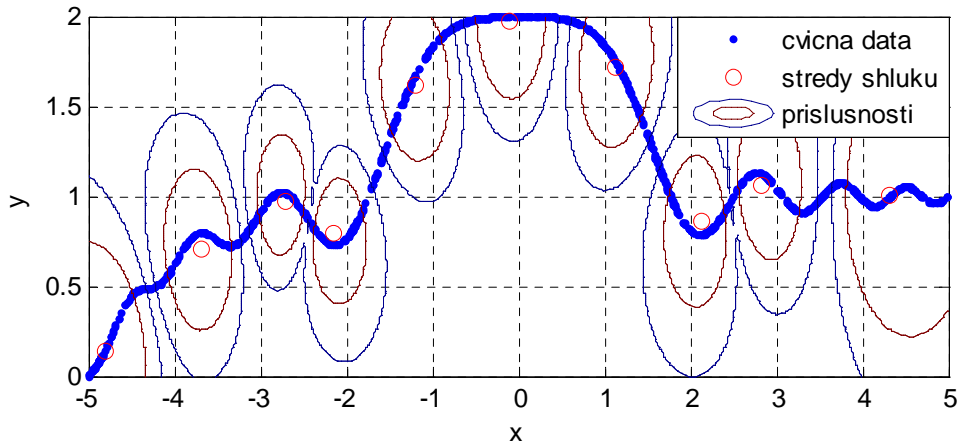


Obr. 4.9.b GK 20 shluků, výsledek aproximace:  $t = 1,5790s$ ,  $RMSE = 0,0105$

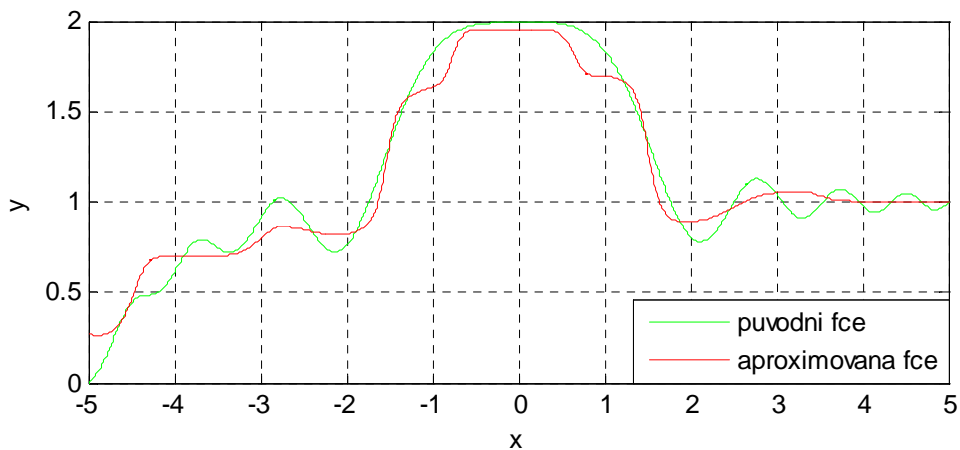


### 4.6.3 Aproximace aperiodické funkce – FCM

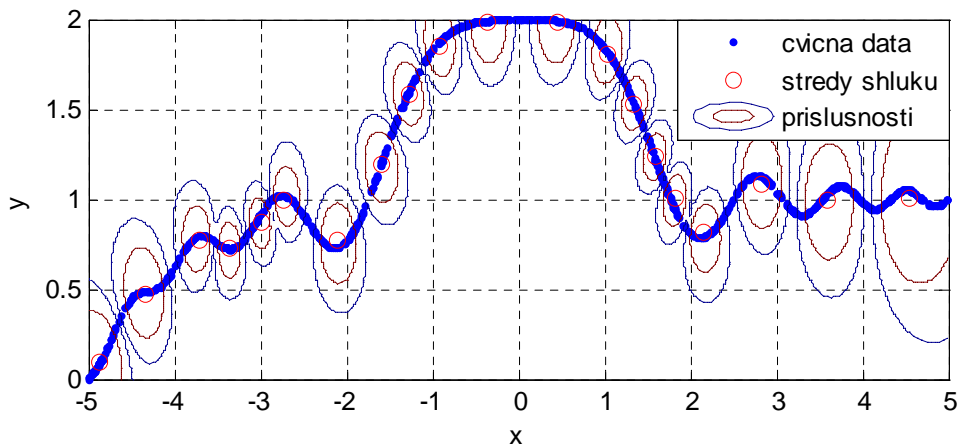
Další měření provedeme pro funkci  $y = 1 - \exp(-x - 5) + \frac{\sin x^2}{x^2}$ . Nejprve opět pro FCM algoritmus.



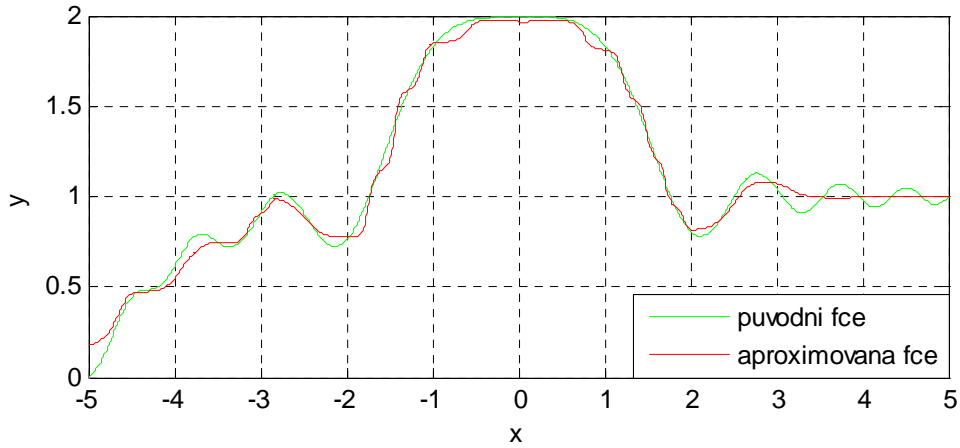
Obr. 4.10.a FCM 10 shluků, cvičná data a nalezené shluky



Obr. 4.10.b FCM 10 shluků, výsledek aproximace:  $t = 0,4680s$ ,  $RMSE = 0,0967$

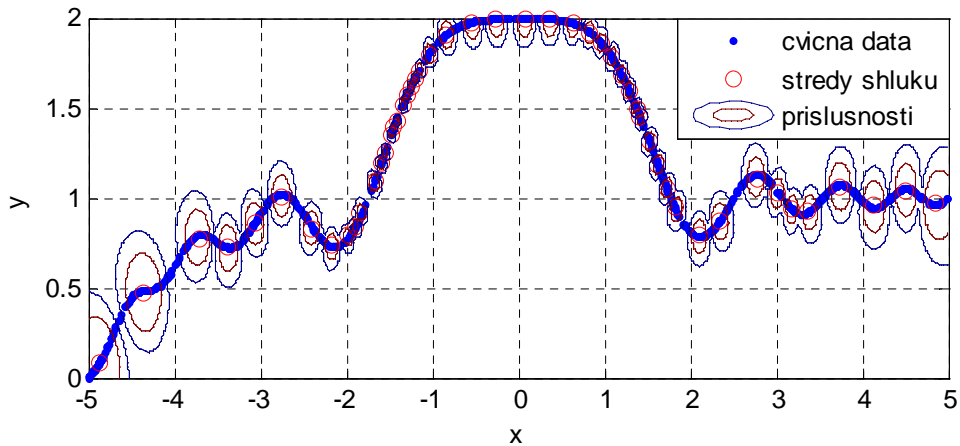


Obr. 4.11.a FCM 20 shluků, cvičná data a nalezené shluky

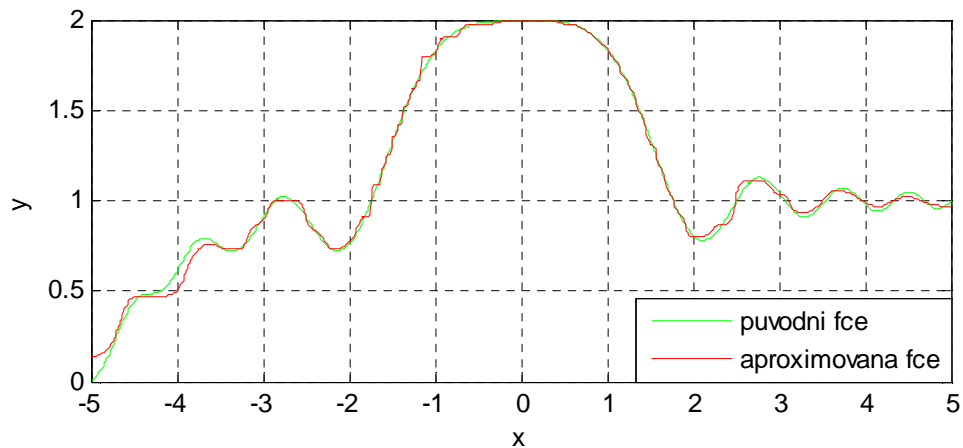


Obr. 4.11.b FCM 20 shluků, výsledek aproximace:  $t = 0,4850s$ ,  $RMSE = 0,0482$

Pro více shluků už nedocházelo k výraznému zlepšení tvaru aproximované funkce. Přesto uvedeme výsledek vyhledání 50 shluků. Vidíme, že algoritmus umístil shluky místo do postranních laloků, kde by to bylo vhodnější, převážně do hlavního laloku.



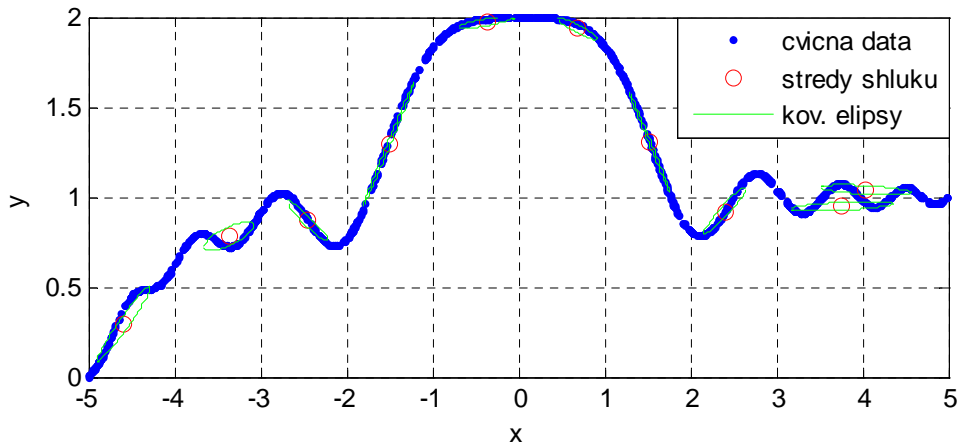
Obr. 4.12.a FCM 50 shluků, cvičná data a nalezené shluky



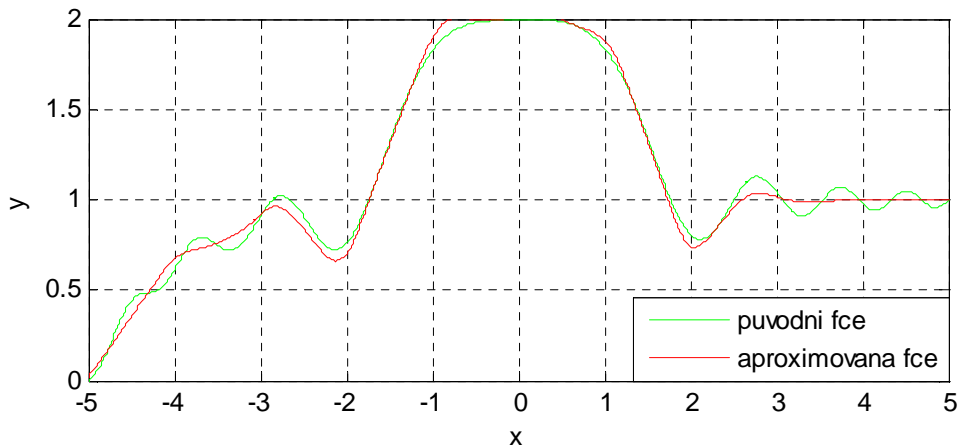
Obr. 4.12.b FCM 50 shluků, výsledek aproximace:  $t = 2,5780s$ ,  $RMSE = 0,0294$

#### 4.6.4 Aproximace aperiodické funkce – GK

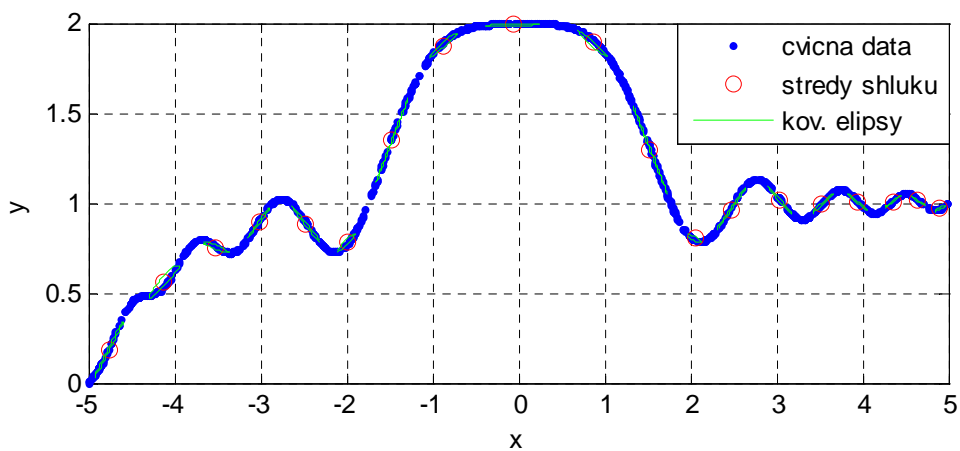
Nyní provedeme měření na stejné funkci, ale pomocí GK algoritmu.



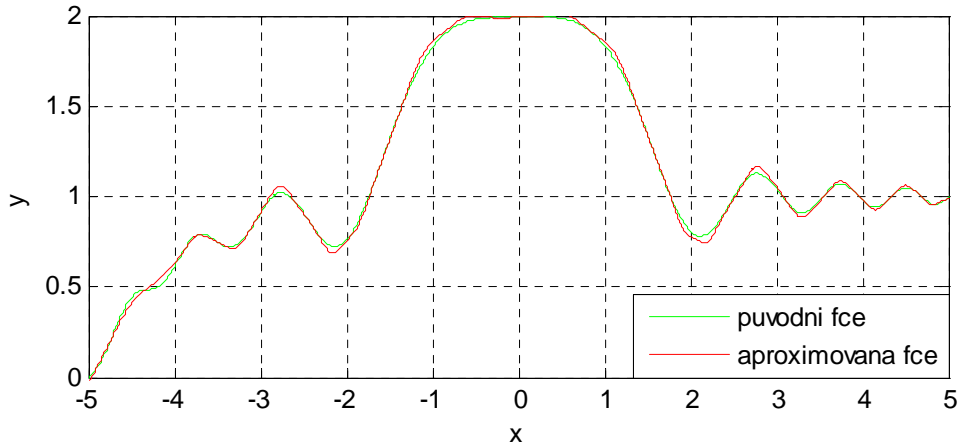
Obr. 4.13.a GK 10 shluků, cvičná data a nalezené shluky



Obr. 4.13.b GK 10 shluků, výsledek aproximace:  $t = 1,9530s$ ,  $RMSE = 0,0468$



Obr. 4.14.a GK 19 shluků, cvičná data a nalezené shluky

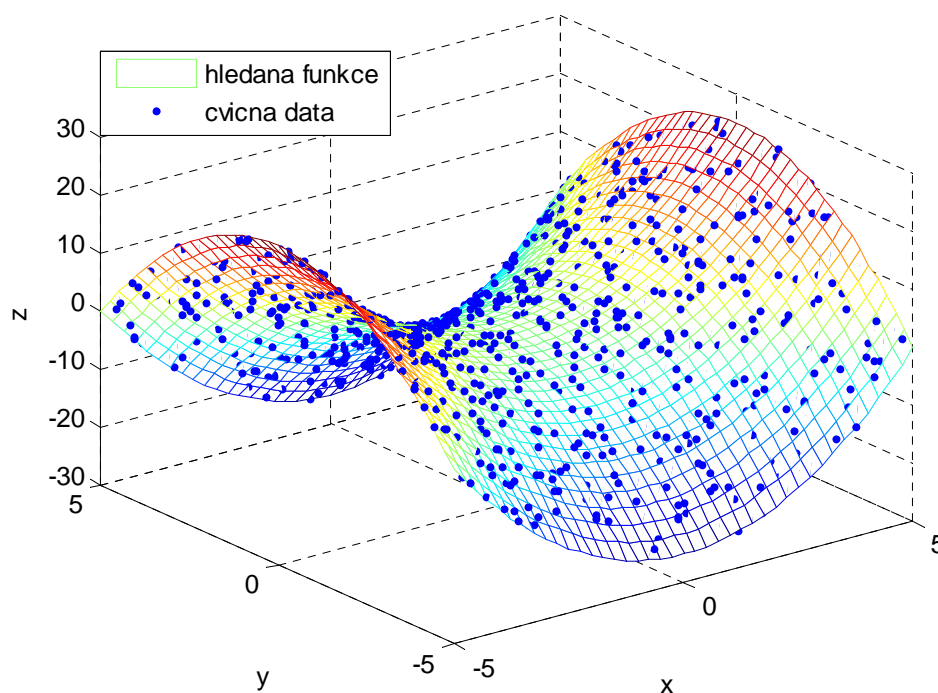


Obr. 4.14.b GK 19 shluků, výsledek aproximace:  $t = 4,7500s$ ,  $RMSE = 0,0182$

Pro 19 shluků jsme dosáhli asi poloviční hodnoty RMSE než pro 20 shluků. Také tvar aproximované funkce byl podobnější původní funkci. Rozdíl byl hlavně v pravé části křivky, kde nebyly proloženy všechny půvlvny. Problém byl v tom, že GK umístil v této části středy shluků do lokálních extrémů. Když se podíváme na obr. 4.14.a, vidíme, že orientace shluků je přibližně ve směru tečny k původní funkci. Jsou-li tedy shluky umístěny v lokálních extrémech, které jsou navíc blízko u sebe, dojde k překryvu shluků a z toho plynoucí špatné aproximaci. Podobný efekt lze vidět na obr. 4.13.a vpravo nahoře.

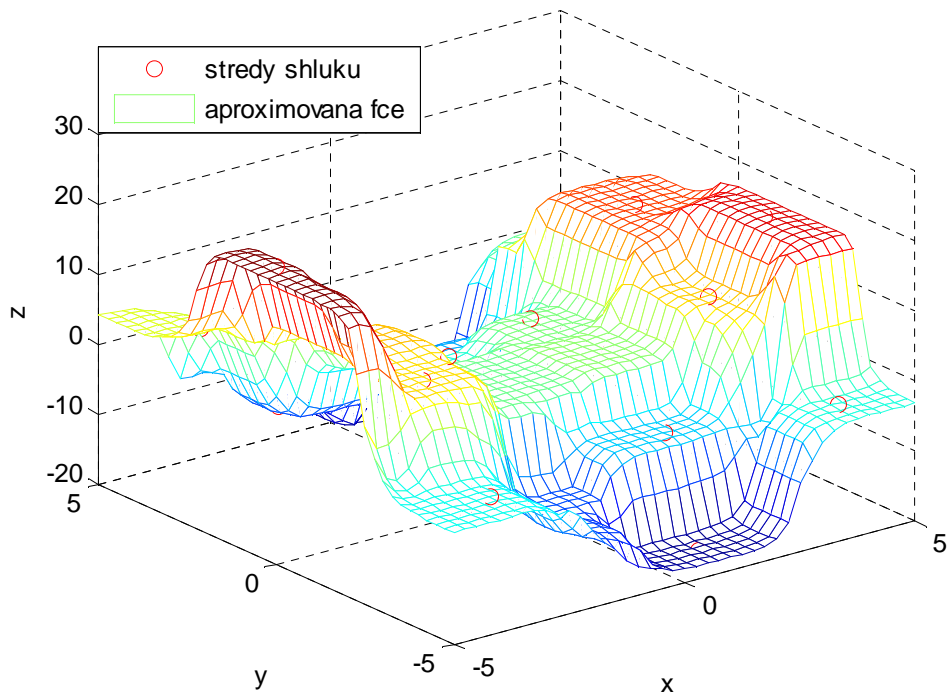
## 4.7 Aproximace funkcí dvou proměnných

V této části budeme porovnávat výsledky aproximace funkcí dvou proměnných. Protože nalezené shluky jsou nyní třídimenzionální, jejich zobrazení by bylo nepřehledné. Oba algoritmy se budou učit na stejné množině dat. Protože nyní nebudeme zobrazovat nalezené shluky, spokojíme se se zobrazením množiny cvičných dat na začátku kapitoly, a to spolu se znázorněním průběhu původní funkce. Pro srovnání opět použijeme směrodatnou odchylku RMSE a pro grafickou představu zobrazíme rozdílovou plochu původní a aproximované funkce. Pro všechna měření při aproximaci funkce dvou proměnných obsahovala množina cvičných dat 900 bodů. Hledaná funkce:  $z = x^2 - y^2$

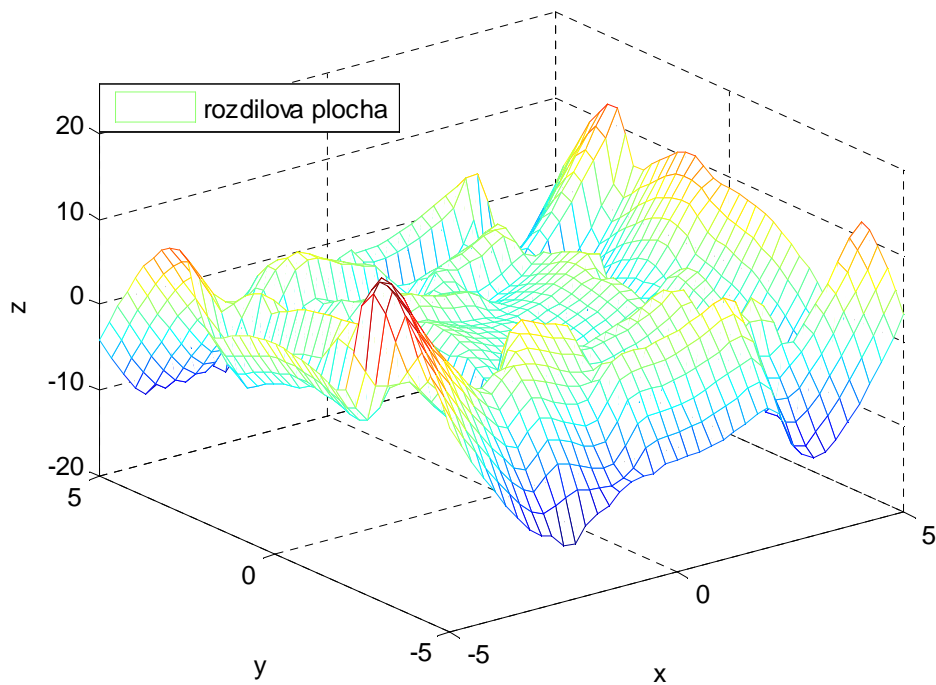


Obr. 4.15 Zobrazení hledané funkce a cvičných dat

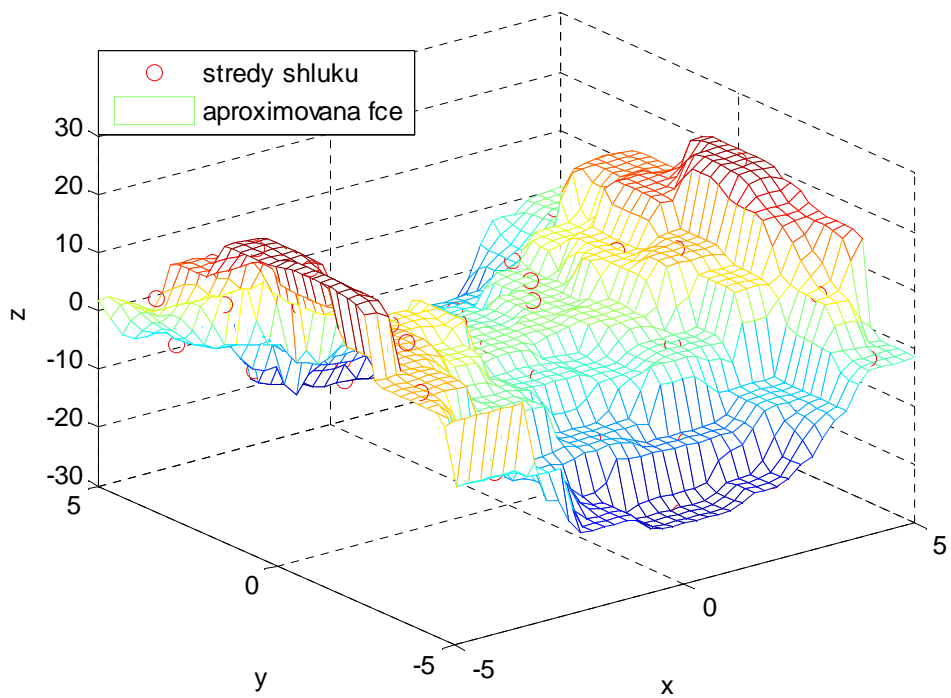
### 4.7.1 Aproximace sedlovité funkce - FCM



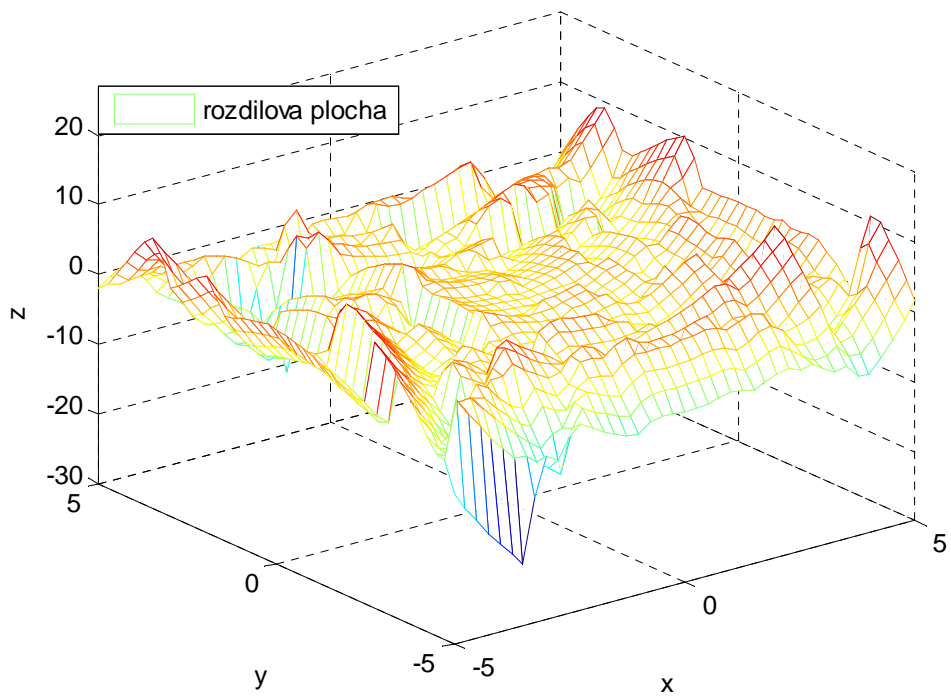
Obr. 4.16.a FCM 20 shluků, výsledek aproximace:  $t = 1,9210s$ ,  $RMSE = 4,4007$



Obr. 4.16.b FCM 20 shluků, rozdílová plocha

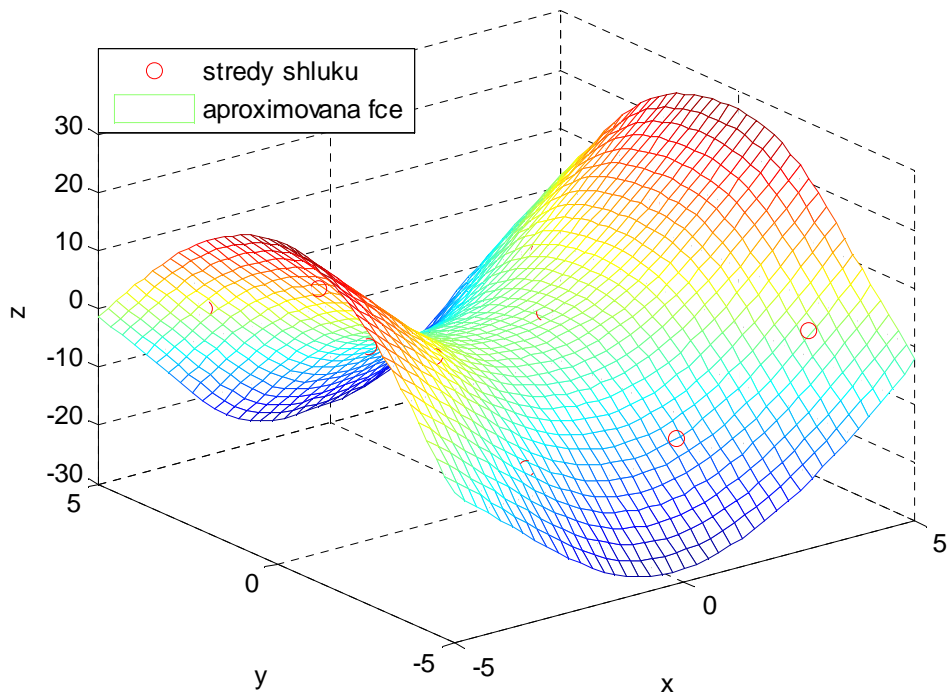


Obr. 4.17.a FCM 50 shluků, výsledek aproximace:  $t = 2,7180s$ ,  $RMSE = 3,1540$

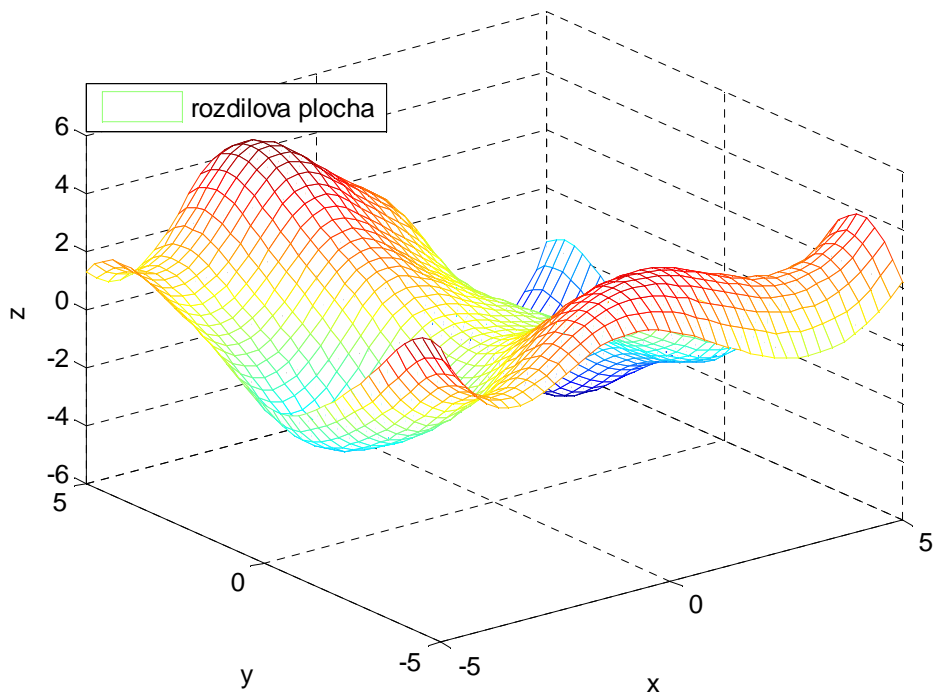


Obr. 4.17.b FCM 50 shluků, rozdílová plocha

## 4.7.2 Aproximace sedlovité funkce - GK

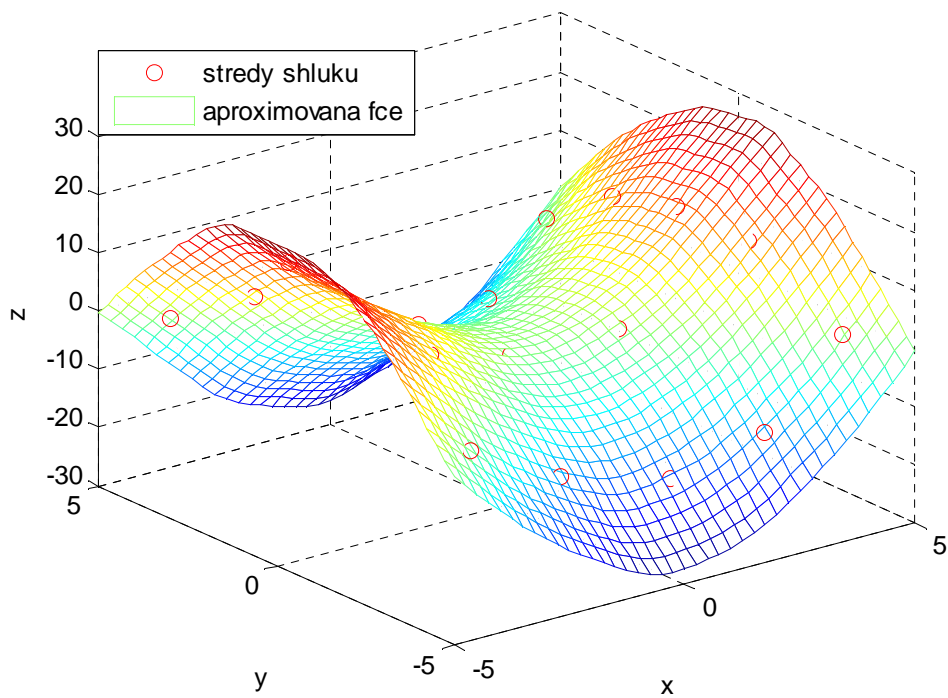


Obr. 4.18.a GK 10 shluků, výsledek aproximace:  $t = 1,3280s$ ,  $RMSE = 2,1909$

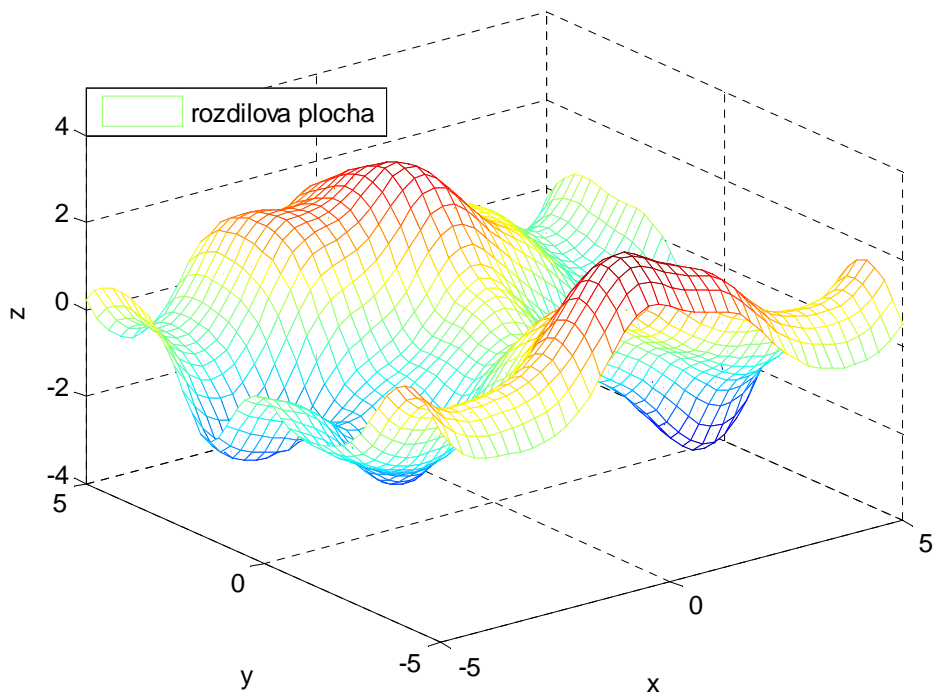


Obr. 4.18.b GK 10 shluků, rozdílová plocha



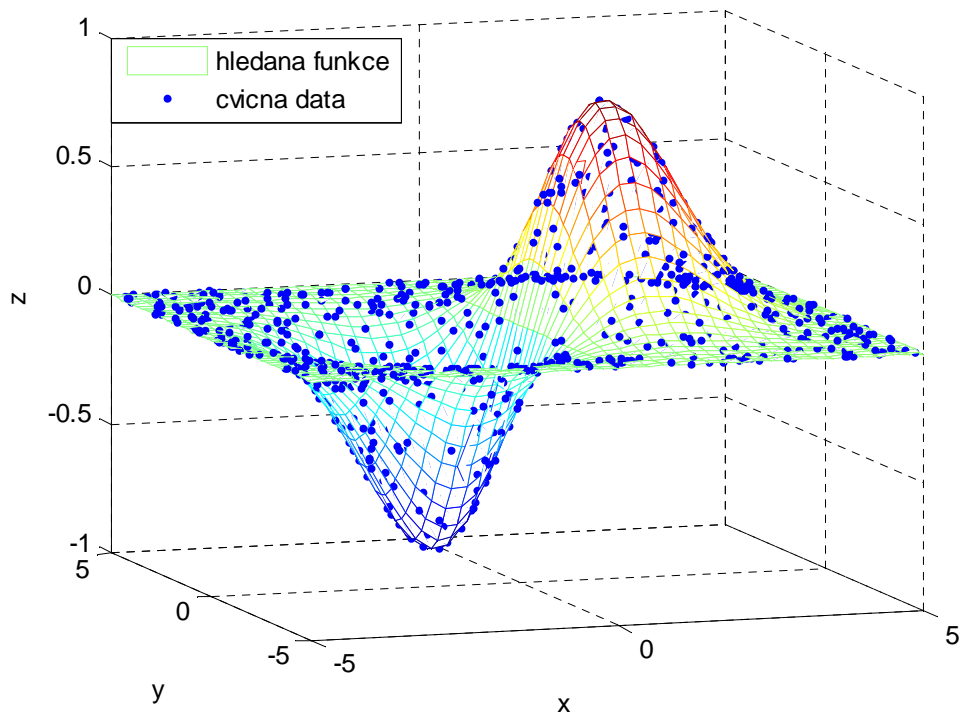


Obr. 4.19.a GK 20 shluků, výsledek aproximace:  $t = 2,0940s$ ,  $RMSE = 1,4454$



Obr. 4.19.b GK 20 shluků, rozdílová plocha

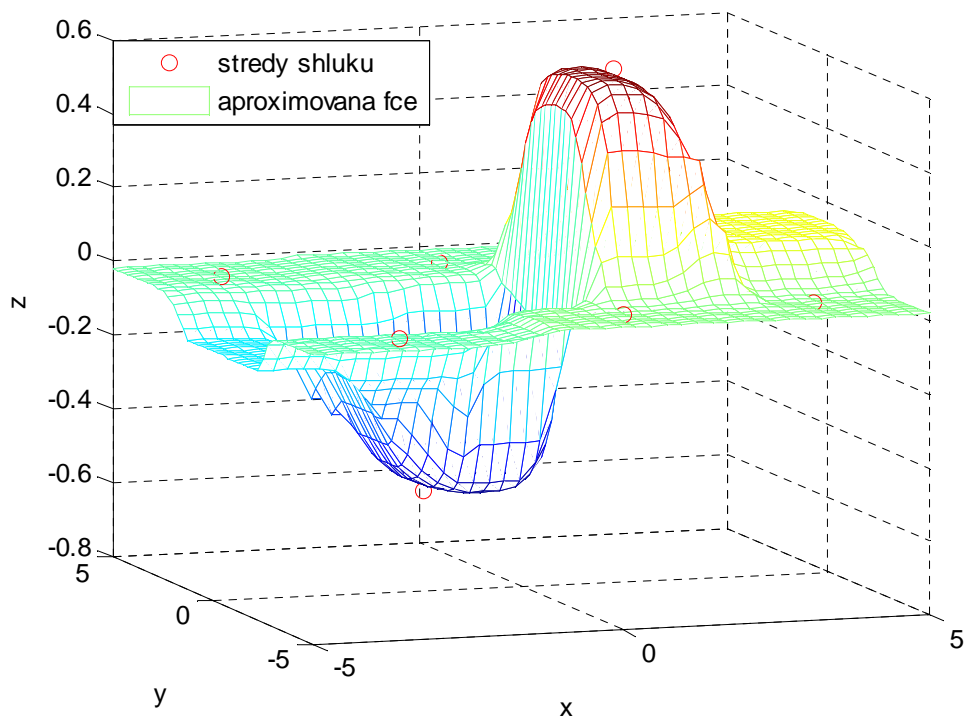
### 4.7.3 Aproximace dvouvrcholové funkce



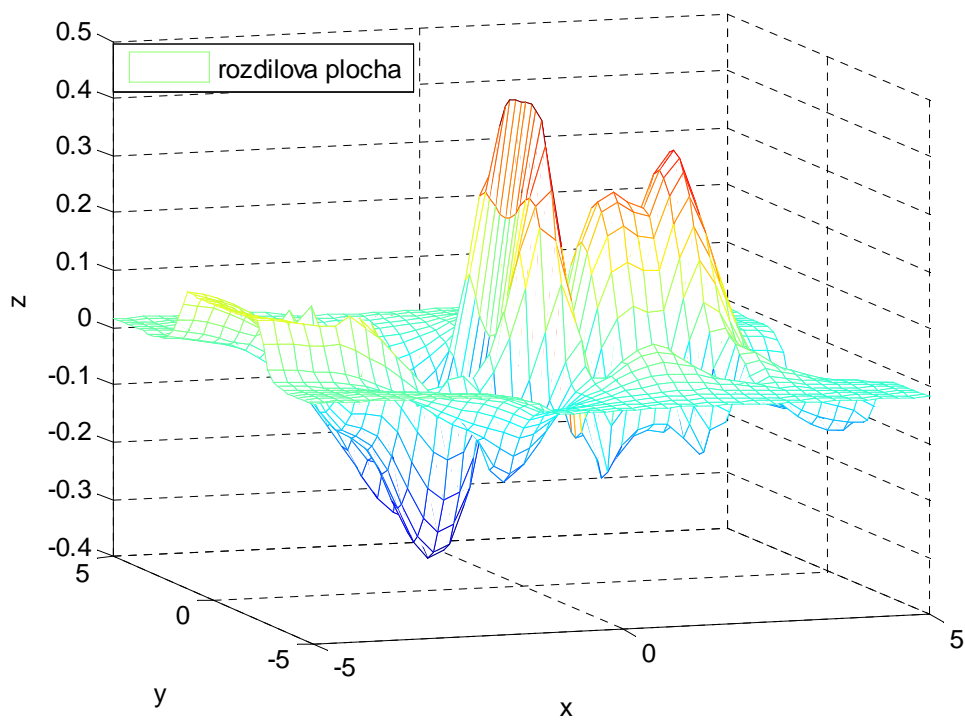
Obr. 4.20 Zobrazení hledané funkce tentokrát bez cvičných dat

Hledaná funkce zobrazená na obr. 4.20 je dána rovnicí  $z = x \exp(-(x^2 + y^2)/4)$ .

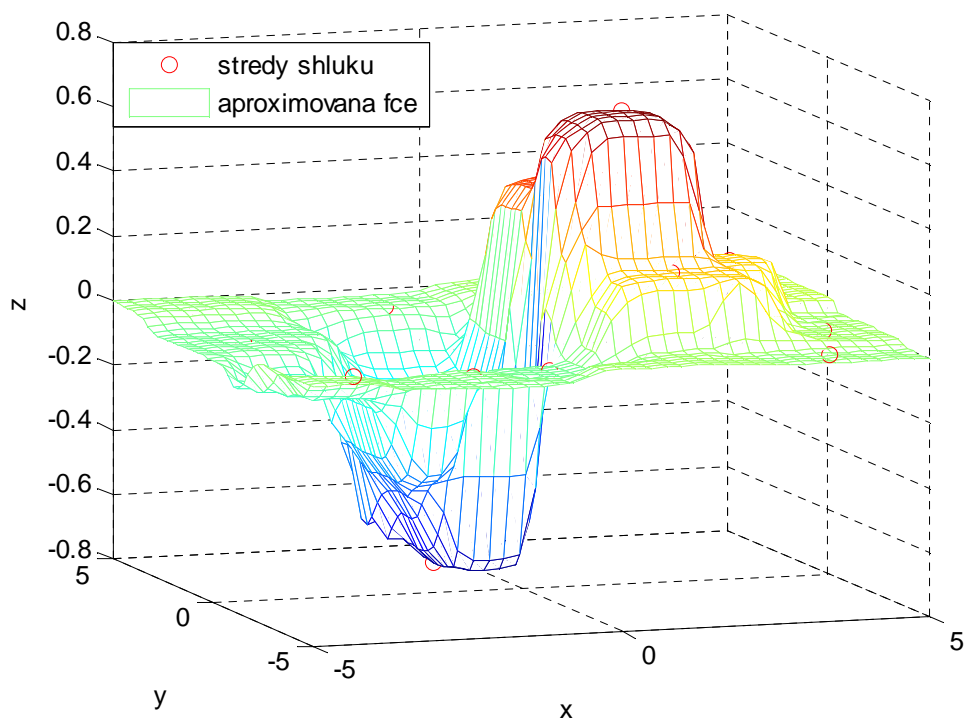
#### 4.7.4 Aproximace dvouvrcholové funkce – FCM



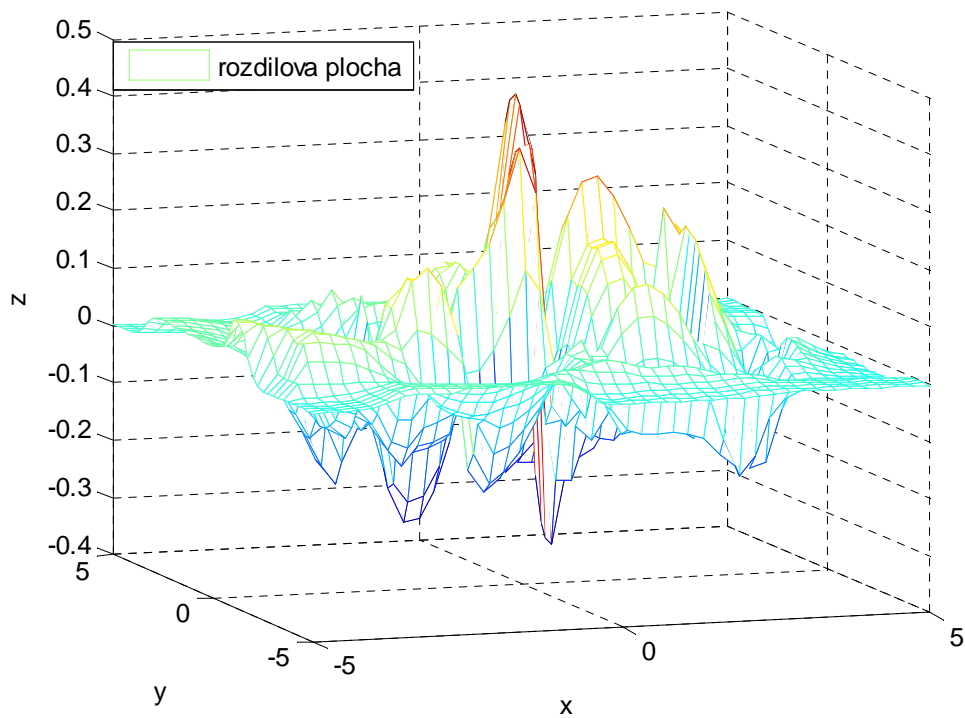
Obr. 4.21.a FCM 10 shluků, výsledek aproximace:  $t = 0,7190s$ ,  $RMSE = 0,1000$



Obr. 4.21.b FCM 10 shluků, rozdílová plocha

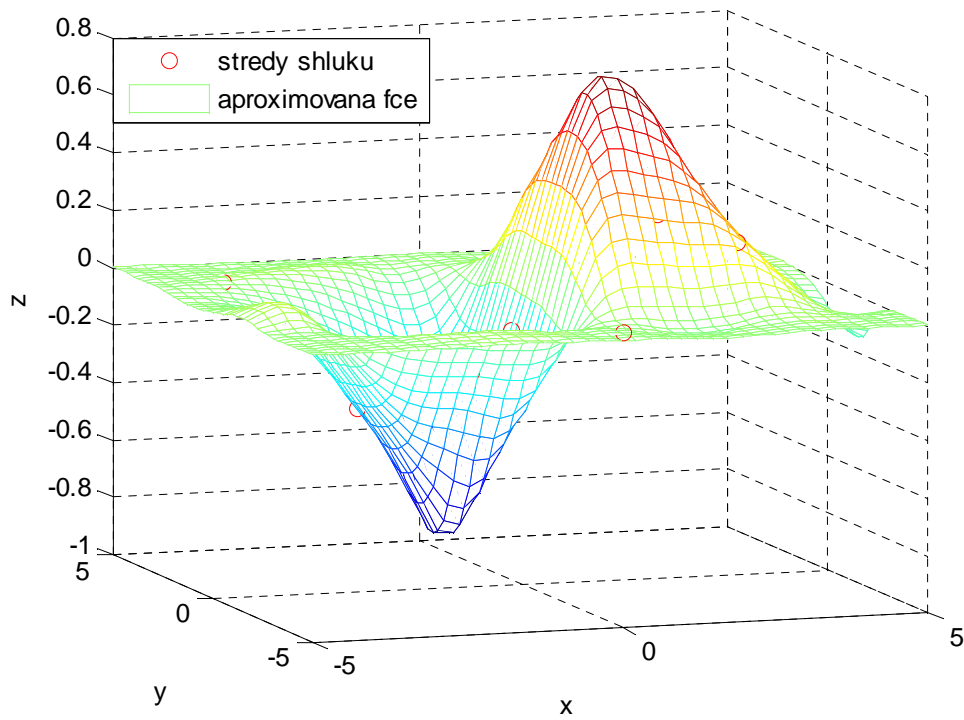


Obr. 4.22.a FCM 20 shluků, výsledek aproximace:  $t = 0,8910s$ ,  $RMSE = 0,0792$

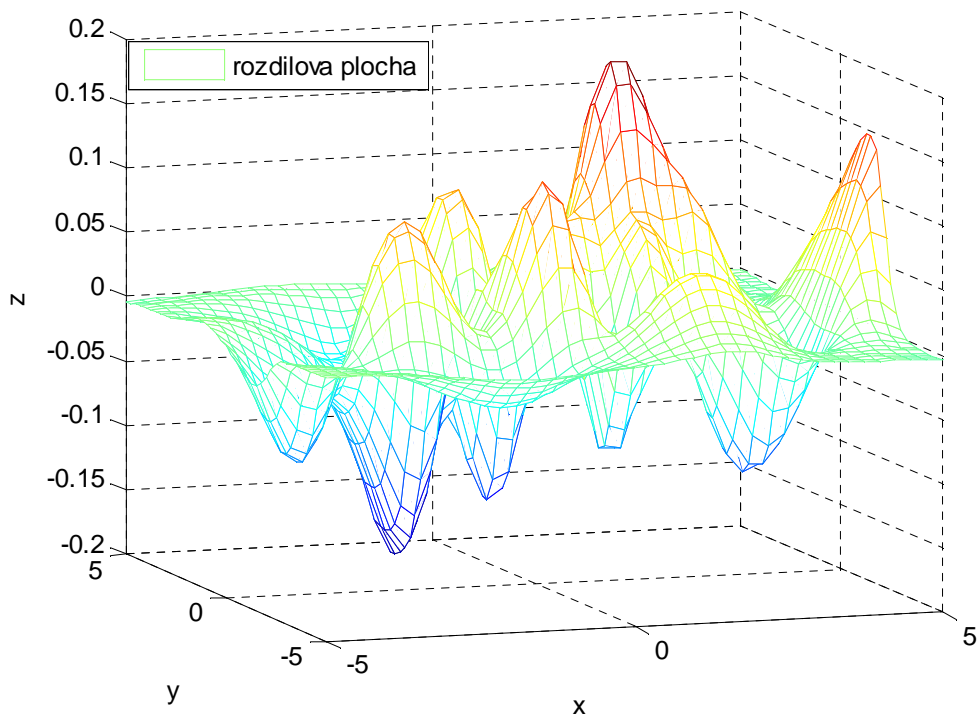


Obr. 4.22.b FCM 20 shluků, rozdílová plocha

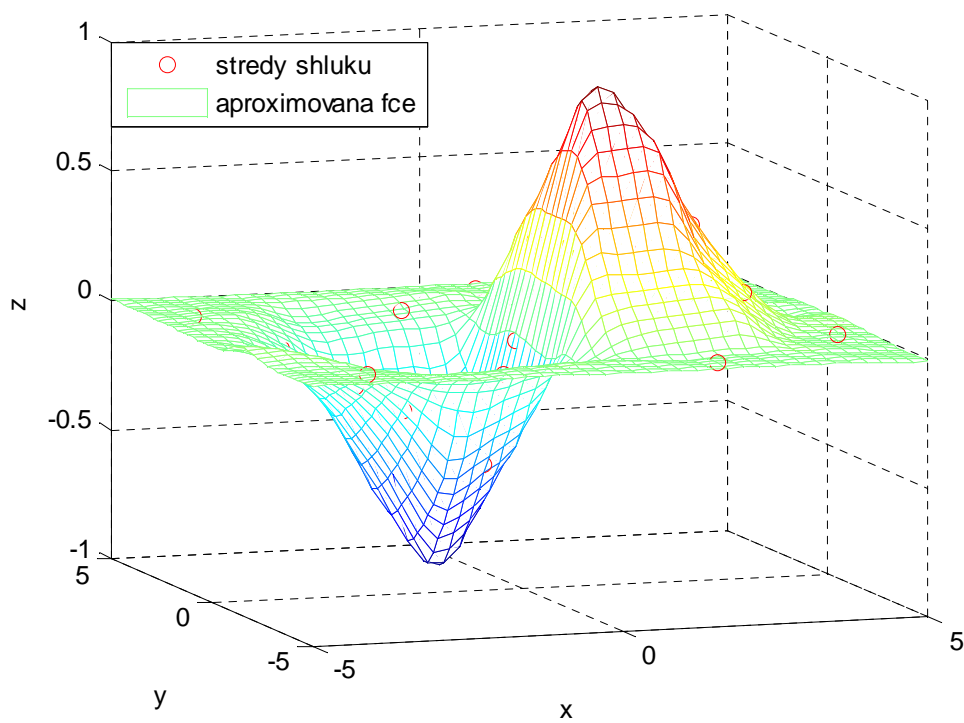
#### 4.7.5 Aproximace dvouvrcholové funkce – GK



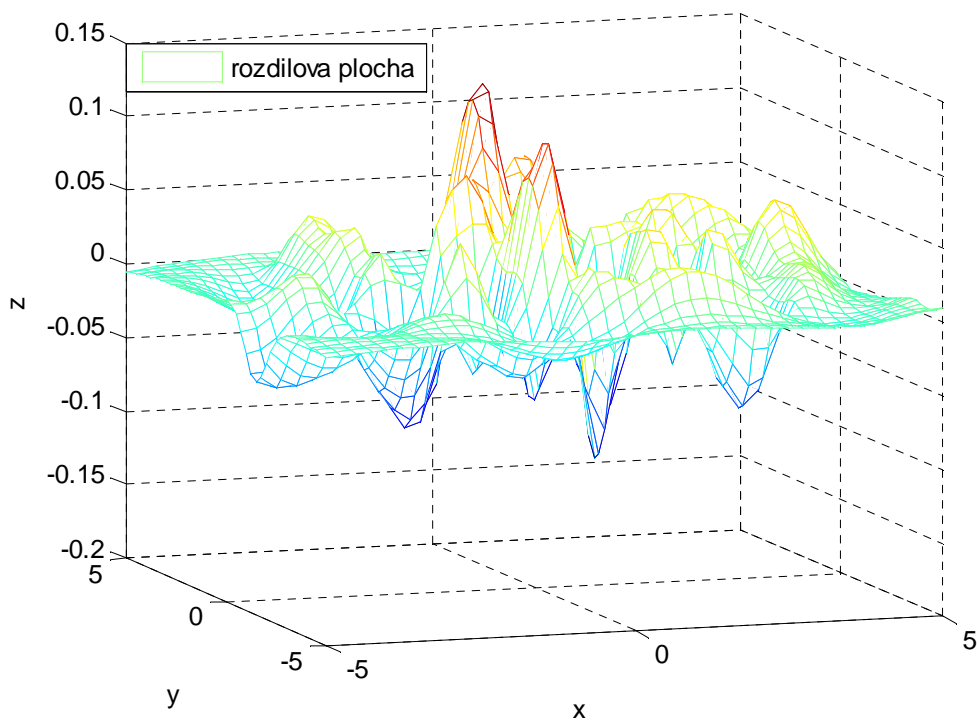
Obr. 4.23.a GK 10 shluků, výsledek aproximace:  $t = 1,6250s$ ,  $RMSE = 0,0486$



. 4.23.b GK 10 shluků, rozdílová plocha



Obr. 4.24.a GK 20 shluků, výsledek aproximace:  $t = 3,2810s$ ,  $RMSE = 0,0265$



4.23.b GK 20 shluků, rozdílová plocha

## 4.8 Zhodnocení výsledků aproximace

V následujících dvou tabulkách jsou uvedeny hodnoty směrodatné odchylky RMSE a času potřebného pro vykonání algoritmů FCM nebo GK. V tabulce 4.1 jsou uspořádány výsledky aproximace funkcí jedné proměnné, v tabulce 4.2 pak funkcí dvou proměnných.

počet shluků	$y = \cos x$				$y = 1 - \exp(-x - 5) + x^{-2} \sin x^2$			
	FCM		GK		FCM		GK	
	t [s]	RMSE [-]	t [s]	RMSE [-]	t [s]	RMSE [-]	t [s]	RMSE [-]
5	0,0780	0,3816	0,1880	1,1031	-	-	-	-
10	0,1410	0,1358	0,6250	0,0275	0,4680	0,0967	1,9530	0,0468
19	-	-	-	-	-	-	4,7500	0,0182
20	2,0310	0,0590	1,5790	0,0105	0,4850	0,0482	-	-
50	-	-	-	-	2,5780	0,0294	-	-

Tab. 4.1 Výsledky aproximace funkcí jedné proměnné

počet shluků	$z = x^2 - y^2$				$z = x \exp\left(-\frac{(x^2 + y^2)}{4}\right)$			
	FCM		GK		FCM		GK	
	t [s]	RMSE [-]	t [s]	RMSE [-]	t [s]	RMSE [-]	t [s]	RMSE [-]
10	-	-	1,3280	2,1909	0,7190	0,1000	1,6250	0,0486
20	1,9210	4,4007	2,0940	1,4454	0,8910	0,0792	3,2810	0,0265
50	2,7180	3,1540	-	-	-	-	-	-

Tab. 4.2 Výsledky aproximace funkcí dvou proměnných

Srovnáme-li výsledky měření časové náročnosti obou algoritmů, vidíme, že FCM je pro stejný počet shluků ve většině případů rychlejší než GK. Odůvodnění spočívá ve vyšší výpočetní složitosti GK, jak již bylo zmíněno v kapitole 2.9. Naopak pro adekvátní počet shluků vykazuje GK lepší (menší) směrodatnou odchylku RMSE než FCM. To je také na první pohled zřejmé ze všech obrázků kapitol 4.6 a 4.7. Lepší aproximace vyplývá z použití Takagi-Sugeno fuzzy systému, jenž obsahuje v závěrové části lineární funkce a ne fuzzy výrok. Výpočetní doba obou algoritmů se zvyšujícím se počtem shluků roste. Na časovou náročnost obou algoritmů má také vliv velikost množiny dat určených k učení algoritmu. Menší množiny dat jsou klasifikovány do shluků rychleji. Pokud bychom tedy chtěli kvůli rychlosti zmenšit množinu dat nebo kvůli zvětšení přesnosti aproximace zvýšit počet shluků, měli bychom si dát pozor na zachování vyváženého poměru mezi množstvím cvičných dat a počtem shluků, abychom nedospěli do takové situace, kdy máme stejný počet shluků jako cvičných dat.

## 5 Závěr

V této práci byla vyzkoušena vhodnost dvou algoritmů fuzzy shlukové analýzy pro aproximaci funkcí jedné a dvou proměnných. Konkrétně byl použit Gustafson-Kesselův algoritmus a fuzzy c-means algoritmus. Hlavní nevýhodou obou algoritmů je fakt, že počet shluků nevypočítává algoritmus sám, ale musí mu být zadán. V této práci není tento problém řešen. Je zde však testována kvalita řešení pro různé počty shluků.

Dále zde bylo vyřešeno nastavení pravidel Takagi-Sugeno a Mamdani fuzzy systémů tak, aby tyto systémy aproximovaly hledanou funkci. Nastavování je prováděno pomocí výsledků algoritmů FCM a GK. U Mamdani systému se jednotlivé části pravidel nastavují pomocí průmětu shluků (nalezených pomocí algoritmu) do jednotlivých os. V případě Takagi-Sugeno systému je navíc nastavena část pravidla pomocí polohy a orientace shluku.

V práci byl ověřen předpoklad, že GK algoritmus spolu s Takagi-Sugeno fuzzy systémem je pro aproximaci funkcí vhodnější, což je dokumentováno na řadě obrázků v kapitole 4. Už pro nízký počet shluků vykazuje dobré výsledky a relativně malé odchylky od hledané funkce. Jeho nevýhodou je vyšší časová náročnost. Oproti tomu je FCM algoritmus rychlejší, ale kombinace jeho výsledků s Mamdani fuzzy systémem přináší uspokojivou aproximaci až při použití vyššího počtu shluků. Přitom musí být zmíněn fakt, že aproximovaná funkce původní průběh nesleduje přesně, ale spíše se kolem něj „vlní“.

Kvalita aproximace byla měřena pomocí směrodatné odchylky RMSE (4.5). Bylo také použito grafické znázornění, které je pro zobrazení funkcí v trojrozměrném prostoru názornější. Pro funkce jedné proměnné je znázorněna aproximovaná a hledaná funkce v jednom obrázku, u funkcí dvou proměnných je kvůli přehlednosti zobrazena rozdílová plocha. Z té lze snadno vyčíst, v kterých místech je aproximace kvalitní a v kterých méně.



### Seznam použité literatury:

- [1] Hoppner F., Klawonn F., Kruse R., Runkler T.; Fuzzy Cluster Analysis, John Wiley & Sons, Ltd., 1999
- [2] Vysoký P.; Fuzzy řízení, Vydavatelství ČVUT, 1996
- [3] Navara M., Olšák P.; Základy fuzzy množin, Nakladatelství ČVUT, 1997
- [4] KMeans; <http://gerstner.felk.cvut.cz/biolab/X33BMI/slides/KMeans.pdf>; duben 2008
- [5] Fuzzy Clustering and Data Analysis Toolbox;  
[www.fmt.vein.hu/softcomp/FuzzyClusteringToolbox.pdf](http://www.fmt.vein.hu/softcomp/FuzzyClusteringToolbox.pdf); březen 2008
- [6] Chuman J.; Modelování nelineárních systémů pomocí algoritmů shlukové analýzy, bakalářská práce, ČVUT 2006